Introduction to Logical Structures

Willem Francois Esterhuyse

1 Department of Mathematics, University of Pretoria, Tshwane, South Africa

IJASR 2021
VOLUME 4
ISSUE 6 NOVEMBER – DECEMBER **ISSN: 2581-7876**

Abstract: We develop symbols for proving Modus Ponens, Modus Tollens and Syllogism more easily. We also develop axioms for these purposes. We do this by postulating certain operators having axioms of application. We state that Logical Structures (LS) requires only one level of abstraction and less symbols than Standard Logic. Standard Logic requires three levels of abstraction to prove the same. The Pilosophical advantage of LS is that it states reasonig clearer, preciser and is closer to written or spoken language than Standard Logic. LS is a second order logic since quantification over enclosures is allowed. The symbols can also be used to construct advanced "Mind maps" of knowledge. This is a major reason for introducing this alternative.

Keywords: Logical Structures, Logic, Modus Ponens, Modus Tollens, Syllogism.

Introduction

Contents:

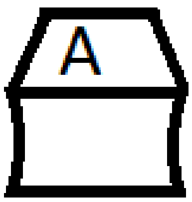
1. Developing the symbols.

Since we are first to develop these symbols, they can be chosen arbitrarially as long as they do not conflict with known symbols.

The symbol for true information follows from the letter "I" as printed and is drawn as follows:

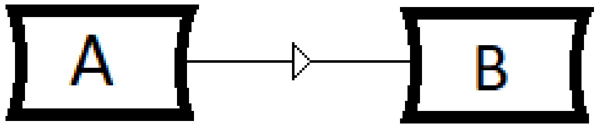
Where "A" is a variable. A true information enclosure may also contain a word or sentence that we want to treat as a unit.

We can also attach a tab to a true information enclosure and label it as:

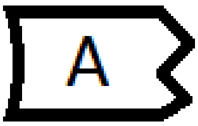


We would refer to this in text as: "Information packet A".

If true information B follows from A we draw it as:

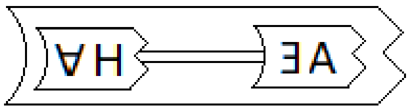


This can also be read as: “A therefore B”.

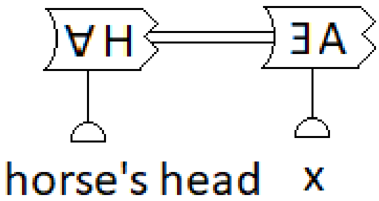


These we call “structures”. An “idea enclosure” will be drawn as:

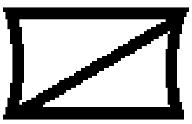
Let H in an idea enclosures denote the idea "horse" and A in an idea enclosure denote the idea "animal". Then we can draw the idea:



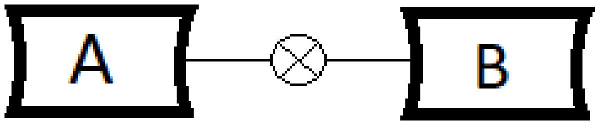
where the first idea in the idea is: "all horses" and the second idea is "some animal". Thus this reads: "The Idea that all horses are some animal". Now we may use the Introductor operator and a variable phrase x to write:



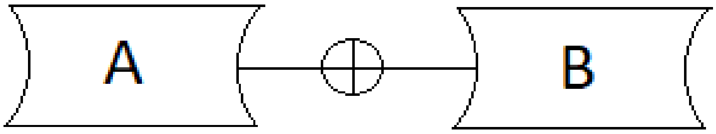
and from this determine that x must read: "animal's head".
Ideas have truth values: "exist" and "don't exist".
An empty information enclosure is drawn as:



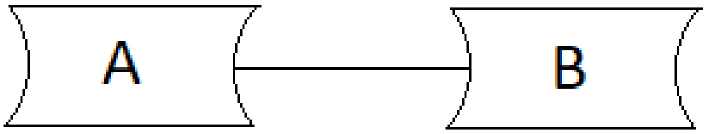
This can be used to specify that some true information enclosure fits here. We will use it for "and introduction". For the statement "information A and information B" we draw:

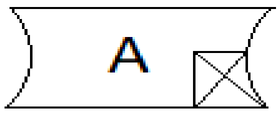


Note that A and B above cannot swap places like in arithmetic since A may preceed B in time. The statement: "information A or information B" is drawn as:

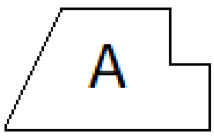


We make the convention that true information will simply be stated as information. The statement "information A is relevant to information B" is drawn as follows:

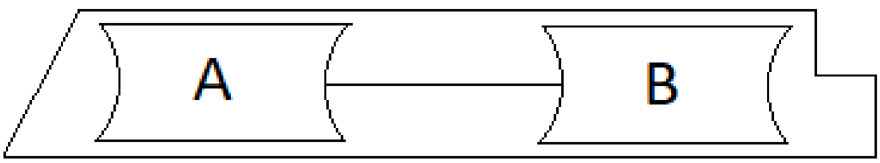




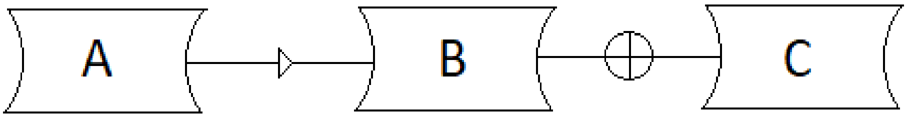
True information is the default and for untrue information we will use:
For a structure in general we use:



The same that applies to information enclosures applies to structure enclosures.
Other enclosures and symbols may fit in a structure enclosure as for example:

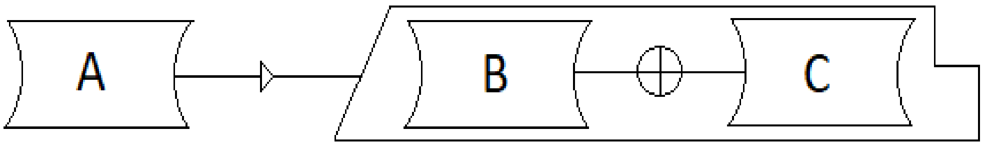


A valid structure is one that has a meaning, expressible in natural language.
It remains to explain how combinations of the above symbols shall be used. For example:



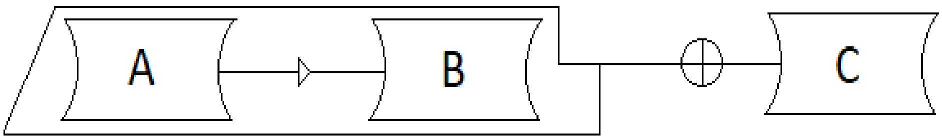
(1)

Will mean:



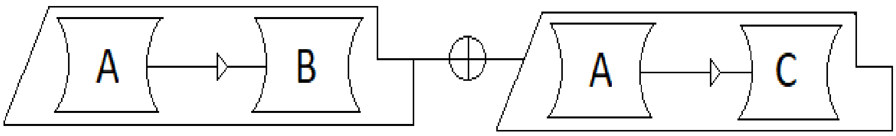
(2)

since the alternative:



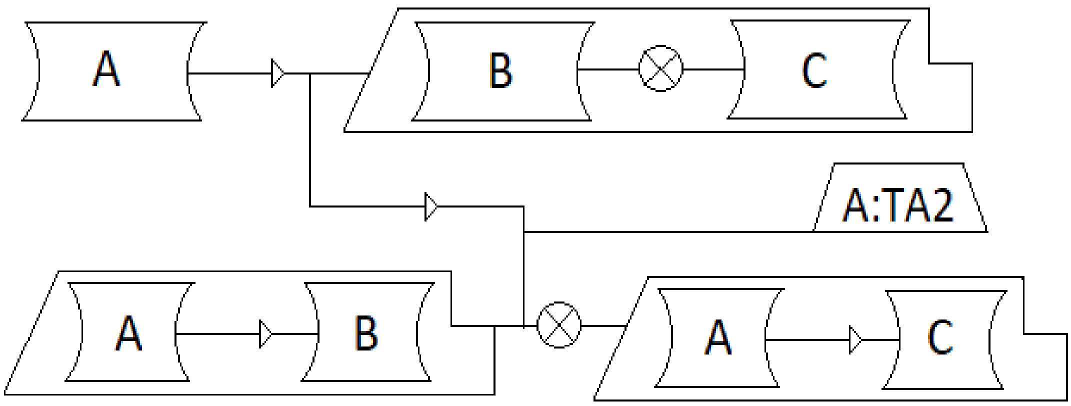
(3)

is not intuitive. This is why (1) is bad practice. We note that (2) would mean:



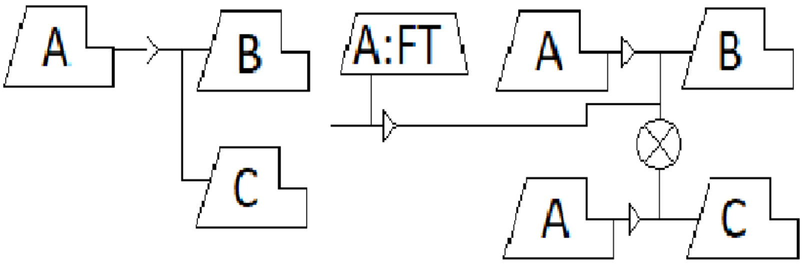
(4)

and (1) therefore (2) would be our first axiom. This reads: "structure of (informtion A therefore information B) or structure of (information A therefore information C)". This is intuitive from (2) but not from (1). (2) therefore (4) would be our second axiom (A:TO2). The same would apply for "and" in place of "or". We will call the axiom so altered: A:TA2:

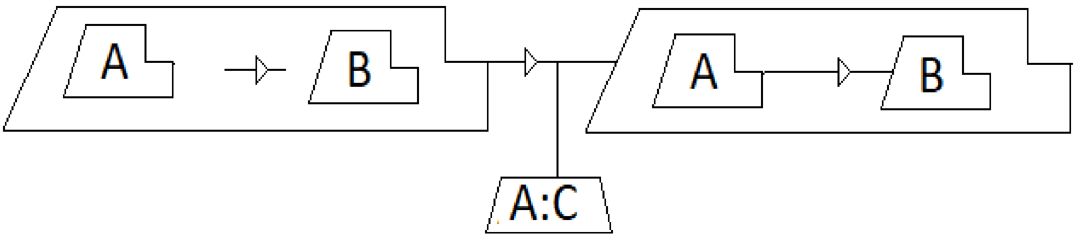


Where the label attach to the main connective. The main connective always occur outside a structure enclosure. These axioms extends to the analogous case of n "ands" in the obvious way.

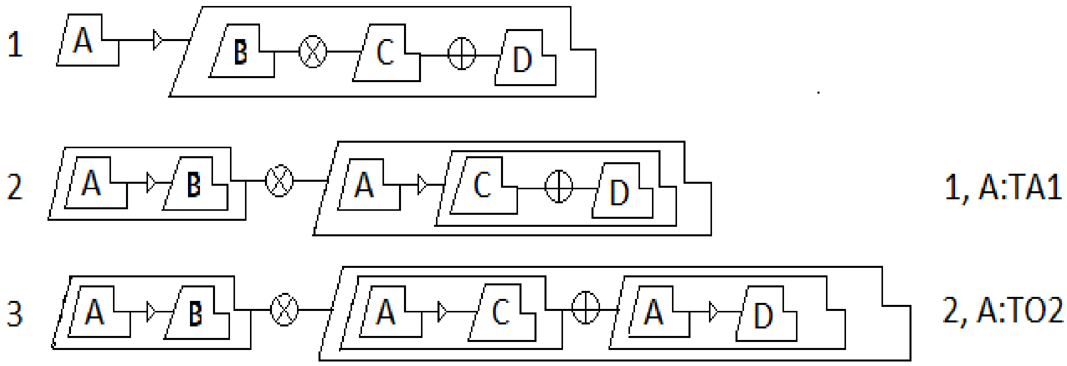
An alternative to drawing "and" is shown in LS of the following axiom:



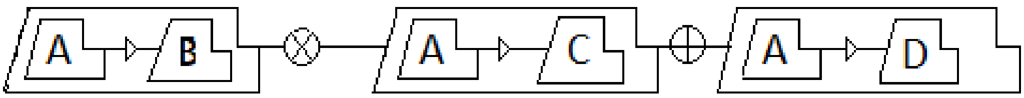
We need another axiom to deal with non-connecting relations:



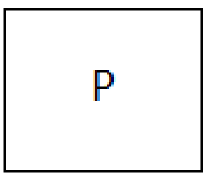
For a mixture of "or" and "and" we derive as follows:



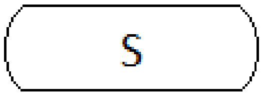
Note that nothing allows us to write this as:



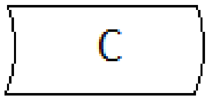
Although it looks reasonable.
A "process enclosure" is drawn as:



It must contain the name of a logical process or another type of process.
A scope enclosure looks like:

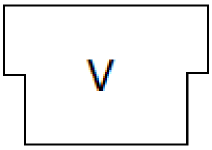


The contents must indicate the subject matter of a structure.
A concept enclosure looks like:



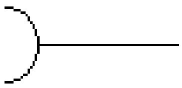
A concept has truth value: "exists and is usefull" and "doesn't exist".

We introduce another enclosure called a Variable Enclosure. It is drawn as:



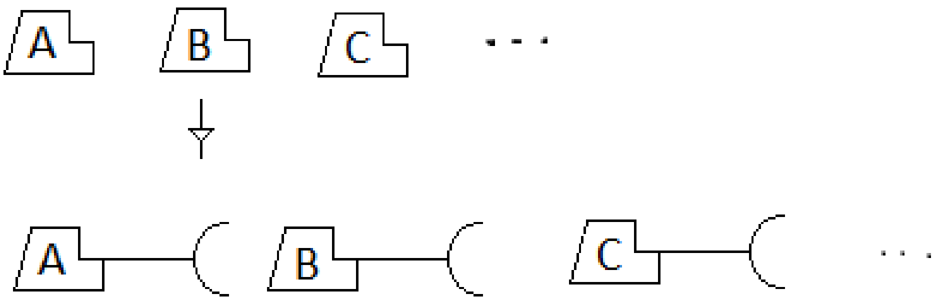
Any enclosure may replace this enclosure in a structure.

For proving MP we need the following operator:



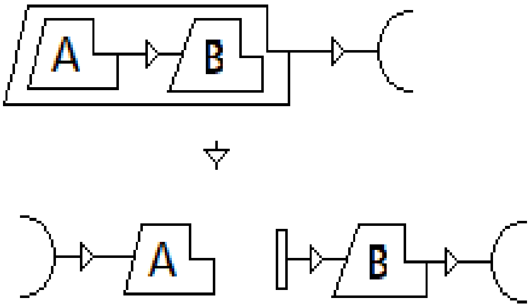
Attractors may carry a relation symbol indicating the relation it may break.

It operates in two ways: Attractor introduction (A:AI) as follows:



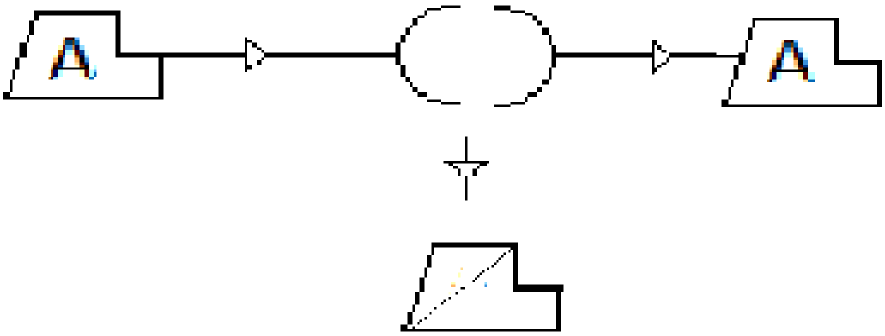
i.e. they are introduced to structures such that they all face in the same direction.

Attractor distribution (A:AD1) works as:

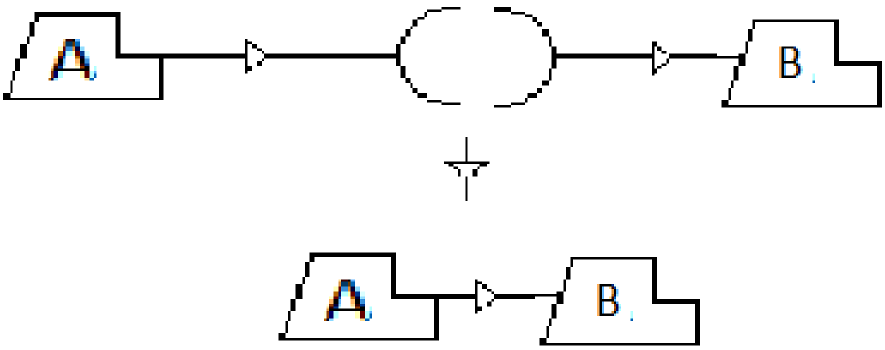


Where the operator in the centre is called a Stopper. It indicates the relation can't reconnect. The Stopper can go on either side of the therefore symbol. Note the direction of the "therefore" symbol.

Annihilation using Attractors (A:AA) goes as:



Attractor Linking (A:AL) is another axiom. It goes as:

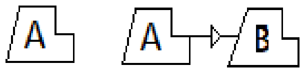


Another axiom is: in a statement all Stoppers may be exchanged with Attractors and vice versa (A:SAE). This happens in one step.

The last axiom is: at either end of a statement Stoppers may be dropped (A:SD).
2. Proving Modus Ponens and Syllogism

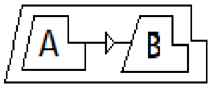
We prove MP using the well known method of writing a proof. The method specify as follows: the line number goes on the left, the statement goes in the middle and the reason for the derivation goes at rightmost.

We begin with the premises:

1 

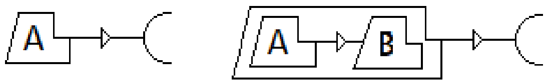
Premise

These are two structures. We make this explicit:

2 

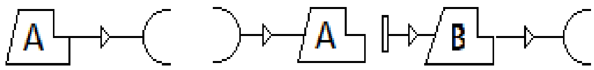
1, rewritten

Now we may introduce Attractors carrying a therefore symbol:

3 

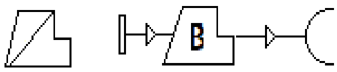
2, A:AI

We apply Attractor distribution to this to get:

4 

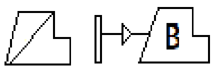
3, A: AD

Because we have matching structures A and the arrow carried by the Attractors point in the same direction the structures A annihilate:

5 

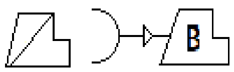
4, A:AA

Because there is no structure to the right of structure B, we may drop the Attractor:

5 

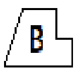
4, A:ADR

Now do Attractor-Stopper exchange:

6 

5, A:ASE

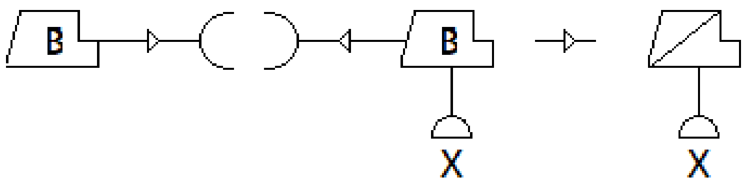
and drop the Attractor, to get:

7 

6, A:ADR

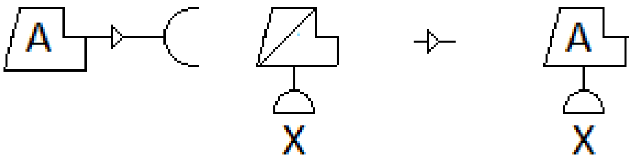
This proves MP.

For Modus Tollens we need the following two axioms:

A:AN 

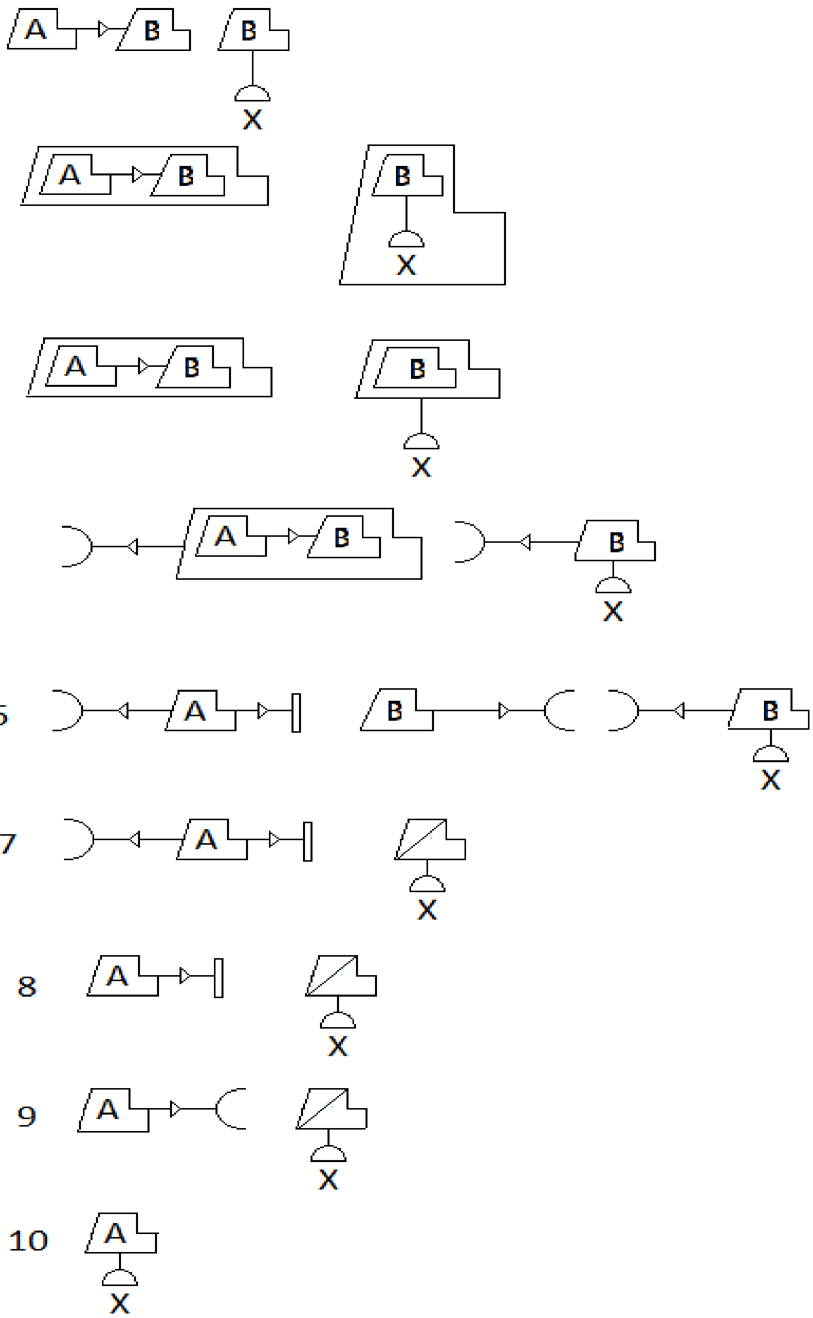
where the operator is called an Introductor and in this case it introduces negation. An Introductor can also introduce words into enclosures, labels into enclosures, enclosures into structure enclosures, structures into structure enclosures, and existential or universal quantification into enclosures. And:

A:NA



Intuitively the Attractor attracts the Introductor.

We prove Modus Tollens:



Premise

2 — 1, restated.

3 — 2, restated.

5 — 3, A:AI

6 — 5, A:AD

7 — 6, A:AN

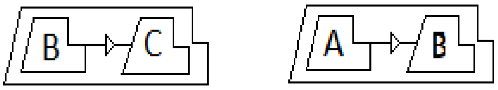
8 — 7, A:ADR

9 — 8, A:ASE

10 — 9, A:NA

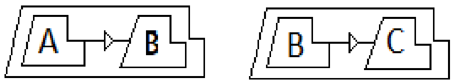
This proves Modus Tollens.

Next we prove Syllogism. The premises are:

1 

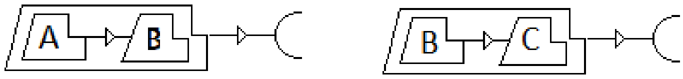
Premises

Rearrange this:

2 

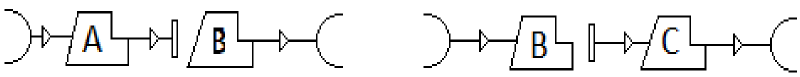
1, rearrange

Now we may introduce Attractors carrying therefore symbols:

3 

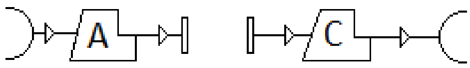
2, A:AI

Distribute the Attractors:

4 

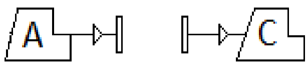
3, A:AD

After Attractor annihilation we get:

5 

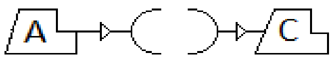
4, A:AA

Dropping the nonsense Attractors we get:

6 

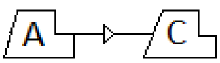
5, A:ADR

Now we do Attractor-Stopper exchange to get:

7 

6, A:ASE

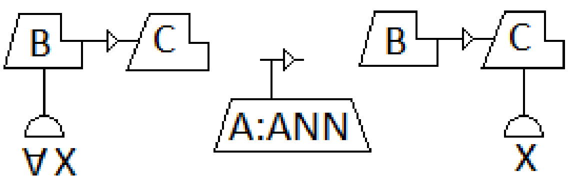
Executing the Attractors we get:

8 

7, A:AE

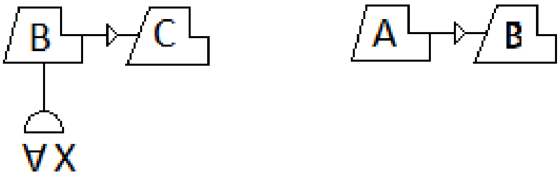
This proves Syllogism. We assume universal quantification if not specified.

Other uses of syllogism (using quantifiers other than universal quantifiers) can be proven similarly, just using introductors for the quantifiers on the relevant structure(s). For this we need another axiom:



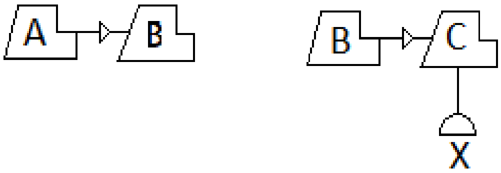
note that "all-not" is not equivalent to "not all".

For example if the premises are (Celarent (EAE-1)):

1 

Premises

we rearange the "none" quantifier to get:

2 

1, A:ANN

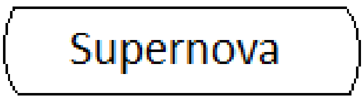
and continue like in the above proof. Notice that the right structure still says "all not" or "all B therefore not-C".

For Darii (AII-1) and the others we do similarly. For Barbari (AAI-1) we just change the universal quantifier into an existential one in the conclusion. I checked that they all can be proven as above. See ref. [2].

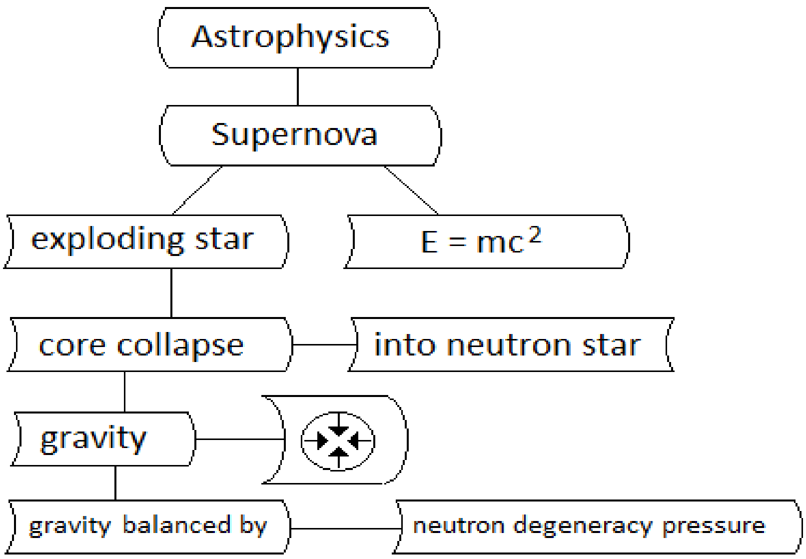
3. Mind Maps.

Mind maps is a way to summerize knowledge in a form that is easily recallable. By constructing mind maps we generate maps of structured information i.e. knowlege.

We start with a Scope Enclosure. For example:

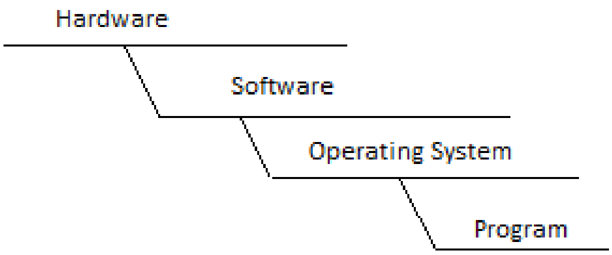


Then continue to add anything related to it as follows:



Note that we use appropriate codes in a standard format as labels. The format is: a letter followed by a colon followed by a three letter code.

One must be able to understand words on lines diagrams as is often encountered in computer books.



where the diagram above reads: Program depends on Operating System depends on Software depends on Hardware.

References:

1 Langer, S.K., An Introduction to Symbolic Logic, Dover Publications, 2011.
2 Wikipedia. https://en.wikipedia.org/wiki/Syllogism#Barbara_(AAA-1) 2020