Raymond Llull's contributions to computer science

John N. Crossley

Clayton School of Information Technology
Monash University, Clayton, Vic., Australia 3800
John.Crossley@infotech.monash.edu.au

This work is scheduled to appear as a chapter in Raimundus Lullus: An Introduction to His Life, Writings and Thought*, ed. Alexander Fidora and Pere Villalba, Turnhout: Brepols (Supplementum Lullianum II), in 2006(?).*

Ramon Llull was a Catalan philosopher, born in Mallorca in 1232, died in Mallorca in 1316(?). His main work, referred to as Ars magna *was a large philosophical treatise proposing a system for raising, in an objective manner, questions about God and the world. In it he developed a new system of logic.*

The relevant Figures *referred to in this report in Section 3 may be found at http://lullianarts.net/Ars-Magna/ars-magna.htm*

Abstract. Llull has been described, anachronistically, as the first computer scientist. Nevertheless he did make significant contributions to the discipline that is now seen as computer science. Some of these contributions are also relevant to mathematics. The latter has become well-known, principally because his ideas were taken up by Leibniz (1646-1716) and his followers. We discuss these contributions. In addition we draw attention to Llull's use of letters for entities and argue that he was the first to use the idea of substituting a value for a variable.

1 Introduction

In 1958 Martin Gardner [10] was the first modern to draw attention to Llull's contributions to logic machines in his book *Logic machines and diagrams* [10]. Llull's contributions to what has become mathematical logic and computer science have been noted over the years. However, assessments have oscillated between brief dismissal and adulation. Thus Prantl dismisses Llull ending a long exposition [20], vol.3, pp.145–177, on p.177 with: 'Dass die ganze Kunst des Lullus schlechthin werthlos ist, bedarf nun wohl keines besonderen Nachweises mehr.' On the other hand Sowa [24], pp.5–6, graciously acknowledges one of Llull's contributions to computer science in that Llull's rotating discs generated large numbers of combinations which could then be tested. Others, such as Bonner [1], pp.65 ff. have regarded Llull as giving rise to the modern theory of graphs but this seems hard to substantiate. Certainly Llull drew interconnections between concepts, but he did not undertake any analysis of the graphs themselves. Llull has been referred to as the 'first computer scientist' in particular by Catalans, but there is also an extensive study by Künzel [12]. Ton Sales [23] describes

Llull as 'One of Us', i.e. a Computer Scientist. Likewise Anthony Bonner [2] asks 'What was Llull up to?' and presents a nice picture from the point of view of a modern computer scientist. However, it is always dangerous to take the concepts of the present day back into medieval times. Llull himself would never, indeed could never, have described himself as a 'computer scientist'. The very phrase, if not the words, was not in circulation.

Nevertheless some ideas of Llull were explicitly used by Leibniz (1646–1716) in [15] (where Llull is mentioned by name several times) and before him by Kircher [11], and led directly into the development of computing machines through Llull's development of a formal language and the mechanics of generating ccombinations. But Llull contributed more than that. He also contributed to the logical notion of a formal language and the mathematical notion of variable and substitution (for a variable).

In this chapter I discuss Llull's contributions. From today's point of view these can be described as follows:

1. the idea of a formal language,
2. the idea of a logical rule,
3. the computation of combinations,
4. the use of binary and ternary relations,
5. the use of symbols for variables,
6. the idea of substitution for a variable,
7. the use of a machine for logic.

Finally I consider the rôle of Llull in relation to the modern disciplines of computer science and computer engineering.

In looking at the contributions that Llull made and which led into computer science it is important to realize three things. First, that the construction of Llull's new logic took place throughout Llull's life. Secondly, that Llull's aim in developing his system was not to develop logic but to provide a means for the conversion of monotheistic non-Christians, specifically Jews and Moslems.[1] Thirdly, the mechanical systems of Llull did not provide answers to the questions. Rather, having posed the question, it would then be possible to use the usual techniques of argument with the result that the Christian's argument would lead to the conversion of the non-Christian. As Colomer [6], p.118, writes: 'El mecanisme lògic de l'Art està al servei d'una finalitat religiosa.' Thus Llull would begin in an objective world and then ask questions. (See [18], p.XVII, and the *Prologue*, p.3, of the *Ars Brevis*.) The objectivity would avoid the Christian taking a superior position and would promote engagement with the interlocutor. The solution of the questions would then convince the hearer of the truth of Christianity.

In order for his system to be effective it was necessary for it to be understood. This seems to have caused Llull considerable difficulty and is, presumably, the reason why he simplified his system over the years. Indeed he spent a very large

[1] This is eloquently expressed by Robert Pring-Mill in [21].

part of his life designing the appropriate (perfect, to use Eco's word in [8]) language.

It seems appropriate to regard Llull's work as being logical in the modern sense. He himself says (*Ars Brevis*, see [18], chapter XII, Form 87):

> *Logica est ars, cum qua logicus invenit naturalem coniunctionem inter subiectum et praedicatum*

In the sections that follow we first recall the modern approach and then view Llull's work from this perspective. We shall principally restrict ourselves to Llull's *Ars Brevis* [18] since the points that we which to make are all included there.

2 The formal language

In modern formal (or symbolic or mathematical) logic the basic expressions of the syntax of the language are built up from primitive elements, sometimes called 'atoms' or 'atomic formulae' by putting them together with the aid of connectives (such as & ('and') or $\rightarrow$ ('implies')). The expressions so obtained are called 'well-formed formulae' or simply 'formulae'. Rules (see below, section 4) may then be used to deduce more formulae. The formulae have interpretations. Thus $p \vee \neg p$, which is read 'p or not p' is interpreted in the obvious way if we have a proposition interpreting p, for example: 'It is raining or it is not raining'.[2]

Llull gave formal procedures for the language that he used. There were a finite number of primitives. Originally he started with many more but in the last analysis these were reduced to nine which form his *alphabet*: B, C, D, E, F, G, H, J, K.[3] These primitives will be interpreted in different ways. We discuss this in detail below in section 7 but for now it suffices to give an example: B is interpreted as goodness, difference, whether?, God, justice or avarice, C as size (or greatness), concordance, what?, angels, prudence, gluttony.

> *B significat bonitatem, differentiam, utrum, Deum, iustitiam et avaritiam.*
> *C significat magnitudinem, concordantiam, quid?, angelum, prudentiam et gulam.*

It should be noted (for section 4 below) that each letter has an interrogative word included in the list.

It is very interesting that he should have opted for a finite number of primitives since Llull's aim is to be able to produce all questions. Later, when we

[2] Here we have moved the position of the 'not' to ease the English syntax. Further, it should be noted that in modern logic the 'or' is the 'inclusive or'.

[3] The letter A has already been used for the name of Llull's first figure. It is interesting that although Llull did not use suggestive notation, dividing the alphabet into different parts for different uses, nevertheless he retained his letterings throughout his work. This explains why his figures are labelled with letters quite remote from each other in the alphabet.

come to combinations we shall see that he was certainly interested in producing large numbers of combinations, though he does not refer to infinitely many being possible. Having determined the primitives he puts these together according to his four *figures*. One can justifiably say that he puts them together according to mechanical rules (see sections 3 and 8 below). These combinations can then be read to give the various possible questions.

2.1 Llull's treatment of language

First we should distinguish between the formulae, namely the combinations of letters, on the one hand and the interpretation of these combinations. One is tempted to read his formulae as simple combinations BC, EK, etc. but this has to be interpreted in the context of the particular figure being used. Secondly, each individual letter is interpreted not only as an abstract noun, e.g. 'bonitas' but also as the corresponding adjective, here 'bonum'. On occasion he goes further and makes verbs out of nouns. This is difficult to render in both Latin and English. Thus in chapter IX when dealing with the fourth subject, which is man, we find:

> *Homo est compositus ex anima et corpore. Ratione cuius deductibilis est per principia et regulas duobus modis, videlicet modo spirituali et modo corporali. Et definitur sic: Homo est animal homificans.*
> *A man is composed of a soul and a body. For this reason he can be studied using the principles and rules in two ways: namely in a spiritual way and in a physical way. And he is defined thus: man is a man-producing animal.*

Following from this second point, we have that Llull, explicitly, interchanges subject and predicate. This is already clear in the case of the first figure which we discuss in the next section, and explicit in the treatment of the third figure in chapter 6.

> *... mutando subiectum in praedicatum*
> *... exchanging subjects with predicates.*

Besides predicates, which traditionally, since Aristotle, took only one argument (to use the modern terminology), Llull introduces binary and ternary relations. However, these are only in the context of his formulae, see Section 6 below.

3 Llull's figures

Llull introduces *figures*, (see `http://lullianarts.net/Ars-Magna/ars-magna.htm`) as he calls them, to yield combinations of letters. Figure A indeed joins primitive elements by making one the subject and one the predicate. Thus 'bonitas' is joined with 'magnum', rather than with 'magnitudo' to give 'Bonitas est magna'.

It is clear from his figure A and the accompanying text that 'bonum' is the neuter form of the adjective, not the abstract noun meaning 'a good thing'. Later we find him creating new verbs from nouns in the ninth part (see below section 4).

Since there is no priority among the letters we can also join 'magnitudo' with 'bonum' to get 'Magnitudo est bona'. This also permits such conjunctions as 'Avaritia est bona' which, since Llull says 'avarice is not good but evil' (Ars generalis ultima, chapter I, 2, shows that these are not true statements, but simply statements, or, as Llull often uses them, questions.

Thus Llull has provided a method for producing such statements. In modern language this is: If x and y are primitive elements (letters of the Alphabet) then x est y is a statement (formula). This form is just like the modern form x and y, or more symbolically $x \,\&\, y$.

The second figure is more complicated to apply. It is denoted by T simply because it came after figure S in earlier versions of Llull's work. The inner triangles each contribute in two ways to the formation of statements. First each side of a triangle connects two letters together; secondly it brings an additional quality into the formula. For example, in the first triangle we have difference, concordance and contrariety. Any one of these will record a type of conjunction between a letter of the alphabet and another one. For example, 'goodness is different from greatness'. We could symbolise this in a modern guise by $\mathit{Diff}(x, y)$. Here Llull is introducing binary relations as opposed to predicates (see Section 6 below).

The third figure continues this process, adding further complexity. This is also where Llull's preoccupation with combinations emerges. The third figure exhibits all the combinations of two letters from the Alphabet. Identical letters, and therefore combinations such as BB, are not permitted. Llull does not regard BC as different from CB and therefore only half of the square matrix is required to represent all possibilities. Hidden under this however is the fact that each letter can also be interpreted according to the interpretation in either the first or second figures, previously discussed.

In the fourth figure the process is continued. Here we genuinely have ternary relations (see also Section 6 below). He produces all the combinations of three letters, where no letter is repeated in the trio.

The details of the interpretation are not spelt out in his treatment of the third figure. This is reserved until he has introduced the definitions and rules. Then he gives the full and explicit interpretation when he describes his Table.

4 Definitions and Rules

The definitions are what we would call axioms. For Llull these are not a chosen set of formulae, but rather particular interpretations of the formulae. They include such statements as

B. Goodness is that whereby good does good.

but although B has the interpretation 'avarice' these 'definitions' do not even refer to avarice. Further it is not clear how the principles are to be chosen. That Llull had a much less clear idea of this is evident from his statement at the end of part 3 in the *Ars generalis ultima*. Here he claims that his principles help to clarify what is going on and also to provide guidance in the way that a statement may be resolved to be true or false.

> *Now there are those who dare to attack our principles with canine fangs and serpentine tongue, as they disparage and slander our definitions. However, the art has principles that mutually help each other, for instance when someone says: "If greatness is the being on account of which goodness is great, then all goodness must be equally great;" this can be refuted with the principles of majority, minority and contrariety which do not allow every kind of goodness to be equally great.*

In a modern system of formal logic, having built the formulae we then have a set of rules by means of which we can deduce (or infer) more formulae. The aim in modern logic is to go from true formulae to true formulae. Llull does indeed have rules, but their purpose is to generate more statements (or questions). However, they do not have the same form as rules of modern logic. Rather they are more like the basic rules for building up formulae. Thus, in modern logic one says, 1. a basic (or atomic) formula is a (well-formed)[4] formula, 2. if A and B are formulae, then $A\&B$ is a formula. In Llull only one iteration of the second step is allowed. Thus the rules are purely syntactic. The semantic considerations are left entirely to the user.

Basically there are ten rules, one associated with each letter, and using the question word associated with that letter.

Again every combination of letters gives rise to many different statements (or questions). Indeed, this process gives rise to a large number of statements (or questions). Not only do we have the various possibilities for each letter, we also have different ways of ordering the letters. He provides a table for the allowed combinations.

5 Llull's Table

The table lists all combinations of three letters from the fourth figure and each is put in a chamber (*camera*). In addition the letter T is included which identifies the particular syntactical construction. So we get a sequence of four letters in each chamber. T may occur in any of the four positions. Of the remaining letters none may be repeated on the same side of the T. Thus CTBD is permitted but CTBB is not.

All of this amounts to forming conjunctions, for example: 'C has B and D', which could be interpreted as 'Greatness has goodness and duration'. The general form of this kind of conjunction is 'x has y and z'.

[4] See Section 2.

Llull has therefore not only gone beyond *predicates* involving only one argument, such as Man(x): x is a man, to *relations* between two elements, such as x is less than y, which are called *binary* relations, and even to *ternary* relations, such as $x + y = z$ or 'John, Mary and Peter constitute a family'. So we can say that here Llull has provided a rule which we could describe as generating formulae involving binary and ternary relations (see Section 6 below).

There is one special restriction. Expressions such as 'C has B and B' are not permitted but 'C has D and C' is. This latter is of the form: 'x has y and x'. Likewise those of the form: 'x has x and y' are also permitted. It should be remembered here that two occurrences of x may be interpreted as an adjective in one place and a noun in another. Thus 'C has D and C' has an interpretation 'Greatness has duration and gluttony'. However it would be misleading to write x_1 has y and x_2 which suggests (to the modern reader) that x_1 and x_2 may be different whereas they are substitutions for the same letter in Llull's alphabet.

When T is the first or last letter, the remaining three letters must be distinct. One rôle of T is to solve a problem familiar to mathematicians. When you see an expression $2+3\times4+5$, how do you evaluate it? The mathematician's approach is to write the expression unambiguously by using brackets. Thus we have variants such as $(2+3)\times(4+5)$ and $2+(3\times4)+5$, with the latter being unambiguous because $2+(3+4)=(2+3)+4$ by the associative law for addition.

In Llull's case when there are three letters to be combined, there may be zero, one or two letters to the left of the T and the other letters at the other side. His use of the letter T indicates the 'bracketing'. Thus BCTB could be written (BC)B in modern notation, whereas BTCB would be written B(CB). The combination TBCB is not permitted, but the combination TBCD puts all of B, C and D on a par. We could regard this as being B & C & D with our usual assumption that the associative law holds for conjunction. That is to say, it does not matter which & we use first.

This bracketing makes the formulae unambiguous. However, the interpretations are many since the letters on the left of the T are interpreted according to the first and those to the right of the T according to the second figure.

Having produced this large number of questions and statements, in his *Ars generalis ultima*, Llull seems to think that he has covered all eventualities. To the modern reader this may be puzzling, since it is clear that the number of questions one might ask is without limit. He has already worked out that there are 36 combinations for the third figure and 252 for the fourth figure – in each case a limited, i.e. finite, number. Nevertheless he says:

This table is a subject in which the intellect achieves universality ...

However, if Llull was able to generate an infinite number of questions, or worse, all questions, then he would be emulating an aspect of God. Perhaps this is why he does not mention the possibility of an unlimited number of questions.

Llull states that there are 252 combinations.[5] In the case of the third figure he was taking combinations of two letters at a time from nine. This gives $9\times8=72$

[5] Ars Brevis, [18], 4th figure. p.22.

combinations. We remove half of these because the order of the *two* letters does not matter. The result is the 36 that Llull has. In the case of three letters we first have $9 \times 8 \times 7 = 504$ combinations. Since it will depend on which side of the letter T two letters occur, we only lose half of these, yielding 252.

6 Relations

It was Bonner in [2] who first pointed out that Llull was using binary and ternary relations. He pointed his readers to the historical notes in [19], p.299-300 where, incidentally, Llull is not mentioned. Llull appears to be the first tointroduce binary and ternary relations as opposed to predicates. Before that time, and certainly in Aristotle, predicates were only unary. Llull does not allow arbitrary binary and tertiary relations. He severely restricts them, but he does give schemata, which therefore means that there are several different relations. Thus in the third figure (see above Section 3, he has two types of binary relation. These differ only in that the order of the arguments is reversed. In Llull's notation this is the difference between the orders BC and CB. In the fourth figure there are ternary relations. These differ in the order of the letters and they are also determined by the location of the letter T. Thus BTCB is very different from BCTB as we have noted above in Section 3. Even more different are BTCD and BCTD where we genuinely have three different letters, but note our comment above in Section 5, about the different interpretations of two occurrences of a letter such as B in a Llullian formula.

All of the relations that Llull introduces are constant relations. He does not introduce variables for relations. Nevertheless, this move frees formal logic from the tyranny of only having one argument. Incidentally, once one has two or more arguments then it is possible to build more complicated relations, for example (in modern notation), $P(x, y)$ & $Q(y, z)$ builds a ternary relation between x, y and z, from two binary ones. Such constructions are impossible with predicates of one argument only. Llull does not go so far, so his sentences correspond to the forms $P(x, y)$ or $R(x, y, z)$ (with P and Q being constant relations) and further combinations of relations are not considered.

7 Substitution

There has been discussion in the literature about the introduction of variables in doing mathematics. Some authors have argued that they were used by Diophantos, early in the Christian era (perhaps *c.* 300AD). However there is a serious question as to whether Diophantos was simply using an abbreviation. He was interested in the solution of an equation (as we would express it today). Therefore there was an unknown. He used a symbol, which seems to be distinct from the letters of the alphabet, for the unknown. This may have been an abbreviation for a word meaning, in essence, 'unknown'. see [7], p.70. However, this symbol did not have a variety of interpretations. It simply stood for a quantity that

it was required to be determined. Other symbols were used for various quantities. Diophantos himself used $\square$ for the square of the unknown, i.e. the number multiplied by itself. This tradition continued through al-Khwarizmi [22] of the ninth century and into the Italian algebraists of the sixteenth century where the unknown was called the *cos* or thing. There was no suggestion that the value of an expression, say one equivalent to $x^2 + 10x$ might change, rather the emphasis was on solving problems equivalent to solving equations such as $x^2 + 10x = 39$. (This, incidentally has the solution $x = 3$ but it also has a negative solution, which was not accepted for a long time, $x = -13$. See [22, 7], p.68.)

We find something rather different in Llull. He does indeed use the letters B, C, D, E, F, G, H, J, K but each of these has *several different interpretations*. This is quite distinct from the earlier uses where the symbol was denoting *the* object, or number, to be found. He has not explicitly said so but he is *substituting* various words for these letters. Here I disagree with Bonner [2] who says 'the letters don't represent variables but constants'. However, there are several *different* constants corresponding to each letter. Moreover Llull has not just one possibility but a variety: six in each case. The full use of variables in algebra was not to emerge until the late sixteenth century in the work of Viète. See especially his *Æquationum recognitione et emendatione*, [25] and also [7] p.99, [5], pp.14–15. In logic it occurred with Leibniz [14] in the seventeenth. In his *In artem analyticam isagoge* Viète did not exploit variables very much but he did use them to denote differing quantities as opposed to Cardano, [4] (also sixteenth century), who only used letters to denote specific unknowns or powers (squares, cubes) of the unknown. For example on p.190 of [25], Viète is using variables as such. Leibniz's work in logic remained unknown for centuries. The Encyclopaedia Britannica [17] suggests that Boole may have been aware of Leibniz's work on symbolic logic but this is not firmly established.[6]

8 Machines

Llull's first two figures had no moving parts. However the fourth figure had two discs independently revolving around a central axis. It therefore seems appropriate to call this a 'machine'. By rotating the discs different alignments between letters (or words) on them yielded, when read, the differing combinations that we have referred to above.

Thus the machines produced results. These results then had to be interpreted (by substituting for various letters) and this resulted in statements. These statements were sometimes true and sometimes false. However the machines did not determine which. It can therefore be said that the machines aided the logic (and the logician).

It is perhaps worth mentioning that the question of input and output from any machine is a separate question from the workings of the machine. This is as true for today's computers as it is for Llull's machines. For a modern machine

[6] The author has found no other acknowledgment of this work of Leibniz before 1900. Boole worked in the nineteenth century.

there has to be an input provided. This is usually done, or at least, stimulated, by punching in various characters on a keyboard. The keyboard is not part of the computation. Likewise when we need an output this is achieved, normally, either by getting an image on a screen or printing something out on paper. Again these are not part of the computation process. (Although, when one purchases a computer, it usually comes with a keyboard and a screen, these two components are actually peripheral to the actual computer and are indeed referred to as 'peripherals', just as a printer is.)

I would therefore argue that it is accurate, and in accordance with modern usage, to say that Llull 'computed results'. But I would also stress that the results that he computed were not the final word. In addition to his computation process he had to provide (or wished the users of his system to provide) the necessary arguments to determine the truth or otherwise of the statements generated from his computations.

The idea of Llull of making more and more complicated combinations was taken up, as is well known, by first Kircher [11, 16] and then Leibniz in his *Ars Combinatoria* [15].

Indeed Leibniz devotes considerable space to Llull. On pp. 39–40 of [15] he is critical, as we were above, of Llull's restriction to nine letters. He asks in particular why Figure and Number are not included among Llull's basic terms. Then Leibniz proceeds to investigate combinations as such, developing his calculations to larger sets of basic elements and imposing conditions on repetitions. This was purely mathematical work. It was only later that Leibniz developed the idea of the *Ars combinatoria* in the sense of being able to resolve logical disputes by taking up pens and calculating. (The famous *Calculamus* quotation is the culmination of this as an idea.)

On pp. 51–52 of Gerhardt's edition of the *Ars combinatoria*, [15], Leibniz discusses the 'wheels' (*rotae*) of Llull and his successors. Later Leibniz was himself to design a machine for computing, in this case, for the four basic arithmetic operations $+, -, \times, \div$ using wheels. There seems to be a much stronger case for calling this a 'computer' than there is for so labelling Llull's machine.

We also note that Leibniz included in his *Ars Combinatoria* a *Demonstratio Existentiae Dei*. This seems to have been added as a separate exercise. It is listed at the very end of Leibniz's list of contents yet occurs at the beginning of the work. This 'proof' does not use combinations at all, but it does use the kind of logical argument that Llull uses to search out the truth of statements produced by his machines. Such an argument goes from axioms in a logical way, quoting which axioms, or previously deduced statements, are being used at each stage of the proof. This clearly foreshadows the rules of formal logic and the style of reasoning that we use today and that were introduced by Boole, Frege and Russell [3, 9, 26]. Leibniz writes on formal logic with a symbolism that is close to today's in his manuscripts [14] and [13]. These works are self-contained and do not in fact refer to Llull, but it is hard to imagine that Leibniz's beginnings of logic in terms of a formal language was not also influenced by Llull.

9 Computer Science and Computer Engineering

Definitions of new disciplines take some time to stabilize and it is certainly not clear that either of the terms 'Computer Science' and 'Computer Engineering' has so far stabilized. 'Computer Science' may be thought of as the science of computing, including the study of the behaviour of computing machines and their capabilities. 'Computer Engineering' centres on the construction of computing machines. Both are inextricably involved in the design and construction of programming languages, the formal languages that we use with computers, though this tends to be more the concern of Computer Science.

Claims have been made that Llull was the first computer scientist, for example, by Sales [23]. It was Martin Gardner [10] who was the first modern to draw attention to Llull in the context of computing. He wisely only refers to Llull making 'logic machines', not 'computers'.

Llull certainly made machines. He also was involved with the design of what is now described as a 'formal language'. The machines did actually compute results. The fact that they were relatively simple results: the combinations of two or three letters (from an alphabet of, in the last analysis, nine), should not detract from Llull's inventiveness.

We agree with Colomer [6], p.132, who writes: 'Hi has primerament una coincidència de caràcter extern, però de cpnseqüències incalculables cara al futur: la formalització del llenguatge, o sigui, la creació d'un llenguatge artificial, en el qual els signes substituteixen les operacions del llenguatge comú.' We therefore conclude that Llull could be described as a 'computer engineer' in that he made logic machines which, after all, was what Turing [27] did last century, but he made significant contributions to what is now know as 'computer science' in that he developed the idea of formal language (formal combinations of symbols), the use of variables and, to a limited extent, binary and ternary relations.

References

1. Anthony Bonner. Ramon Llull: Relació, acció, combinatòria i lògica moderna. *Studia Lulliana*, 34:51–74, 1994.
2. Anthony Bonner. What was Llull up to? In Miquel Bertran and Teodor Rus, editors, *Transformation-Based Reactive Systems Development, 4th International AMAST Workshop on Real-Time Systems and Concurrent and Distributed Software, ARTS'97 Palma, Mallorca, Spain, May 21-23, 1997*, Lecture Notes in Computer Science, pages 1–14. Springer, 1997. `http://www.geocities.com/Athens/Forum/5284/compbon.html`, accessed 10.08.2005.
3. George Boole. *An investigation of the laws of thought on which are founded the mathematical theories of logic and probabilities*. Dover, New York, 1958. Originally published in 1854.
4. Girolamo Cardano. *The Great Art or The Rules of Algebra (Ars magna)*. MIT Press, Cambridge, Mass., 1968. Translated by T.R.Witmer, with a foreword by Oystein Ore.

5. Giovanna Cleonice Cifoletti. *Mathematics and rhetoric: Peletier and Gosselin and the making of the French algebraic tradition*. PhD thesis, Princeton University, N.J., U.S.A., 1992.
6. Eusebi Colomer i Pous, S.J. De Ramon Llull a la moderna informàtica. *Estudios Lulianos*, 23(68–69):113–135, 1979.
7. John Newsome Crossley. *The Emergence of Number*. World Scientific, Singapore, 2nd edition, 1987. First edition, UDA Book Company, Steel's Creek, Australia, 1980.
8. Umberto Eco. *The search for the perfect language*. Blackwell, Oxford, 1995. Translated by James Fentress.
9. Gottlob Frege. *Begriffsschrift und andere Aufsätze*. Olms, Hildesheim, 2nd edition, 1995. Notes by E.Husserl and H.Scholz, edited by I.Angelelli.
10. Martin Gardner. *Logic machines and diagrams*. McGraw-Hill, New York, 1958. 2nd edition, Harvester Press, Brighton, 1983.
11. Athanasius Kircher. *Ars magna sciendi*. J. Jansson, Amsterdam, 1669.
12. Werner Künzel and Heiko Cornelius. *Die Ars Generalis Ultima des Raymundus Lullus, Studien zu einem geheimen Ursprung der Computertheorie*. Berlin, 5th edition, 1991.
13. Gottfried Wilhelm Leibniz. Phil., VII, B, ii 70–71: Essais de Calcul logique. In Louis Couturat, editor, *Opuscules et fragments inédits de Leibniz, Extraits des manuscrits de la Bibliothèque royale de Hanovre*, pages 264 – 269. Olms, Hildesheim, 1961. Reprint of the 1890 edition.
14. Gottfried Wilhelm Leibniz. XIX. [Non inelegans specimen demonstrandi in abstractis]. In C. I. Gerhardt, editor, *Die philosophische Schriften von Gottfried Wilhelm Leibniz*, volume VII, pages 228 – 235. Olms, Hildesheim, 1961. Reprint of the 1890 Berlin edition.
15. Gottfried Wilhelm Leibniz. Dissertatio de Arte Combinatoria. In C. I. Gerhardt, editor, *G. W. Leibniz: Mathematische Schriften*, volume V, pages 7 – 79. Olms, Hildesheim, New York, 1971.
16. Thomas Leinkauf. *Mundus combinatus. Studien zur Struktur der barocken Universalwissenshaft am Beispiel Athanasius Kircher SJ (1602-1680)*. Akademie Verlag, Berlin, 1993.
17. Ramon Llull. Article from *Encyclopdia Britannica Online*, `http://search.eb.com/eb/article-9048651`. Accessed 8.08.2005.
18. Raimundus Lullus. *Ars brevis, Lateinisch–deutsch*. Felix Meiner, Hamburg, 1999. Übersetzt, mit einer Einführung herausgegeben von Alexander Fidora.
19. Anil Nerode and Richard Shore. *Logic for Applications*. Graduate texts in computer science. Springer-Verlag, New York, 2nd edition, 1997. References are to the first edition of 1993.
20. Carl Prantl. *Geschichte der Logik im Abendlande*. Akademische Druck- und Verlagsanstalt, Graz, 1955. 4 vols. in 3: vols. 1, 3 and 4 reprint of the 1855-70 ed., vol.2 reprint of the 2nd ed., 1885.
21. Robert Duguid Forrest Pring-Mill. The analogical structure of the lullian art. In Samuel Miklos Stern, Albert Habib Hourani, and Vivian Brown, editors, *Islamic philosophy and the classical tradition : essays presented by his friends and pupils to Richard Walzer on his seventieth birthday*, pages 315–326. University of South Carolina Press, Columbia, SC, 1973.
22. Frederick Rosen. *Muhammad ibn Musa al-Khwarizmi: Algebra*. Oriental Translation Fund, London, 1831. Reprinted by Olms, 1986.

23. Ton Sales. Llull as computer scientist or why Llull was one of us. In Miquel Bertran and Teodor Rus, editors, *Transformation-Based Reactive Systems Development, 4th International AMAST Workshop on Real-Time Systems and Concurrent and Distributed Software, ARTS'97 Palma, Mallorca, Spain, May 21-23, 1997*, Lecture Notes in Computer Science, pages 15–21. Springer, 1997. `http://www.geocities.com/llull_brazil/compsale.html`, accessed 10.08.2005.
24. John F. Sowa. *Knowledge Representation: Logical, Philosophical and Computational Foundations*. Brooks/Cole, Pacific Grove, CA, 2000.
25. François Viète. Æquationum recognitione et emendatione. In Fracnciscus à Schooten, editor, *Opera mathematica*, pages 82 – 158. Elzevier, Leiden, 1646. Foreword and index by Joseph E. Hofmann, Reprinted Olms, Hildesheim, 1970.
26. Alfred North Whitehead and Bertrand Arthur William Russell. *Principia Mathematica*. Cambridge University Press, Cambridge, 1925–27. 3 vols. First published in 1910–13.
27. David Yates. *Turing's Legacy: A History of Computing at the National Physical Laboratory, 1945–1995*. Michigan State University Press, East Lansing, MI, 1997.