Using Computers to Make Logic Relevant

FREDERICK SUPPE

1. The Standard Conception of Logic Courses

Charles Peirce, in his architechtonic of philosophy, placed logic as a sub-branch of ethics: For did not logic tell us which arguments are good, hence instruct us in the virtues of argument? As two recent authors put the point starting in their logic text:

> Logic is concerned with arguments, good and bad. With the docile and the reasonable, arguments are sometimes useful in settling disputes. With the reasonable, this utility attaches only to good arguments. It is the logician's business to serve the reasonable. Therefore, in the realm of arguments, it is he who distinguishes good from bad.
>
> Virtue among arguments is known as validity.[1]

In principle I am willing to accede to such lofty claims—that it is the logician who distinguishes good from bad in argumentation. But principle is not practice, and practically speaking the logician has precious little to offer in evaluating genuinely problematic arguments.

Nevertheless, the typical logic-text authors pass off logic as a valuable practical means for evaluating arguments. For example, Irving Copi in his *Symbolic Logic* says,

> [t]he study of logic, especially symbolic logic, will tend to increase one's proficiency in reasoning. And . . . the study of logic will give the student certain techniques for testing the validity of *all* arguments, including his own.[2]

Upon reading this the student thinks he is on the threshold of realizing Leibniz' dream, a *Calculus Ratiocinator*, which mechanically will enable him to determine when Nixon is cogent, and whether to be convinced by his Mother's pleas for chastity. But such hopes quickly fade as, instead of such important arguments, his attention is diverted to *P*'s and *Q*'s and the most exciting arguments he evaluates are at best as interesting as

Teaching Philosophy Today, 2nd edition

pp. 111–122

> [n]o student who fails some course that Rudolf teaches fails all the courses that Alfred teaches. Some student fails all the courses and also fails all the courses that Rudolf teaches. Therefore, if Rudolf teaches any courses, then there is a course that every student fails.[3]

And as the semester progresses he feels increasingly cheated.

Is our student wrong? Has he been cheated? Are the textbook authors, and the instructors who transmit their offerings, dealing in fraud? Or is our student just confused—the only failing of instructors in logic being that we haven't succeeded in convincing him that logic (the logic *we* transmit) is of inestimable practical value to him? My own opinion, which I will attempt to defend below, is that our student has the goods on us and on the authors we subsidize by teaching such a course.

If I am correct about this, then the insult to the student often is double. For not only do we present a fraudulently offered course, but often we sanctimoniously require that he take the course (at least if he is to use philosophy to satisfy his humanities liberal arts requirements): After all, is not correct reasoning essential to good philosophy? And how can he appreciate philosophy if he can't reason correctly? And how can he reason correctly if he doesn't know logic?

Before entering into a consideration whether the above charges are justified, let me make a crucial observation: whether the above student reaction is defensible or not, it is a fact of current academic life that this is a typical reaction to introductory logic courses—especially ones which concentrate on symbolic logic. And whether or not we think it a fair or rational reaction, it poses a problem for the teaching of elementary logic courses—namely, how do we make them relevant *in the eyes of our students*? Whether or not it is true that the content of elementary logic courses is in principle an aid to improving the rationality of practical argumentation, on a large scale our courses and the textbooks we use fail both to convince our students this is so and to give ours students any rational argumentative edge in practical disputes. If you don't believe this, just consult typical student evaluations of introductory logic courses taught on the above premise. As an example, I quote from summary student evaluations of such courses taught at the University of Illinois:

> Students were generally disappointed with this course. It is taught basically as a math course dealing entirely with manipulating symbols in a context-free language and proves to have little relevance. "This course was a waste of time except for the fact it fulfilled my LAS [humanities distribution] requirement. It was completely dull, boring, and irrelevant."[4]

Admitting all the above, it is possible for one to react as follows: the problem is not with logic—it's just the way it is taught. Most logic courses are mistaught and fail to realize their promise—which promise is aptly captured in the first two quotations above. My own response to this is that if you escape the above student objections while teaching introductory logic on the foregoing conception, you've done nothing to vindicate that conception of an introductory logic course: You've shown your own charisma, not the viability of such a logic course.

The problem here is not the teaching: it's that formal logic is not a practical means for evaluating genuinely problematic arguments. If you don't believe me, try the following experiment: Take an Agatha Christie short story and, using the techniques of an introductory logic course, try to figure out exactly what Poirot's argument is in solving the case. ("The Kidnapping of Johnny Waverly" is a particularly apt one; I once had a class do this and they understood less how Poirot solved the mystery after doing so than they did when they first read the story). But enough of polemics and examples. I think a kind of theoretical argument can be given which tends to show the practical uselessness of formal logic in evaluating arguments. To adapt an example from Benson Mates (whose skepticism on the practical value of logic is heartwarming but errs in the direction of optimism.),[5] consider the following genuinely problematic argument:

1) Socrates is human.
2) Human is human.
3) For every *A, B, C*: If *A* is *C* and *B* is *C*, then *A* resembles *B* with respect to being *C*.
4) Therefore, Socrates resembles Human with respect to being human.[6]

Is it sound? Since our students are aware (from our courses) that 'is' sometimes signifies identity and other times predication, they interpret 'is' in each instance in what seems the most natural way, using 'is' as identity in the second premise and 'is' predicatively in the other three sentences, and then mechanically determine the argument is invalid although the premises seem true. Does this settle the case? No. for might not the author have used 'is' systematically with just one of these two meanings? If he did, the 'is' of identity is ruled out as being implausible; and, so we take it as the 'is' of predication. Doing so we find the argument becomes valid, but that the second premise is false or meaningless. Well, we've exhausted the standard textbook moves, and so we conclude the argument is unsound. But, notice the following: if we interpret 'is' by the phrase 'is included in' it turns out that the premises are true and the argument is valid; so it *is* sound!

What do we make from this? The following, I think. The argument in question is genuinely problematic as to its soundness. To apply formal logic to evaluate even its validity, we have to resort to a fair degree of philosophical analysis. And depending on our analysis, the application of formalism would be question-begging. And, in this argument, at least, once the analysis has been made, no real question of the validity or soundness of the argument remains to be solved by resort to formalism.

This example makes, I think, the following two points: (1) if the argument is relatively simple (as the above is) yet genuinely problematical, any resort to formal logic will be question-begging unless the formalism is bolstered by an adequate prior philosophical analysis, and the analysis usually will render the formalism superfluous; (2) if the argument is not relatively simple, the formalism may help, but only when it is justified by an adequate prior philosophical analysis of the argument. Therefore (3), the legitimate recourse to logical formalism in the evaluation of genuinely problematic arguments requires a prior philosophical analysis. Moreover (4), formal logic cannot provide that prior philosophical analysis (else via (1) and (2) we get into a vicious infinite regress). From this I conclude that logic itself (at least as measured by the content of the typical introductory logic course) cannot provide an effective practical means for evaluating genuinely problematic arguments.

The argument just given leads to another argument displaying the limits of formal logic for evaluating genuinely problematic arguments. For any attempt to use logical formalism to evaluate *that* argument *ipso facto* will beg the very question!

Thus, our disgruntled student is not "too blind to see the value of what we are teaching" him; rather he has seen through our fraud. And this poses a genuine problem: Why are we teaching him logic under the above fraudulent conception?

2. Other Conceptions of Logic Courses

Instinctively feeling fraudulence in the standard conception of logic courses, some instructors have attempted other approaches to the subject matter.

One reaction to the argumentative irrelevance of standard logic courses is to blame the problems on symbolic logic—perhaps echoing Strawson on its deviation from the use of connectives and quantifiers in ordinary English.[7] typically this takes the form of retreating to the traditional logic of the syllogism and sorites. Not only does this seem to me intellectually dishonest (being akin to teaching only Aristotle's physics in a survey physics course), but it does nothing to save the above concept of logic. For to use a standard example, it demonstrably fails to be adequate for the evaluation of rational arguments, hence is inadequate for evaluating most arguments. It cannot even show sound the following argument; "All horses are animals. Therefore, all heads of horses are heads of animals." At best, traditional logic has an antiquarian interest which properly is relegated to a segment of a course in the history of (ancient and medieval) philosophy.

An obvious approach is to just teach symbolic logic in its full mathematical glory, disclaiming any relevance to it at all. Taken to its extreme, one teaches it as a kind of openly irrelevant intellectual puzzle; this is an approach I've often taken and find many students responsive to. As much merit as this approach has, it leaves unanswered the question of why we should require anyone to take it or make it the typically required portion of a philosophy liberal arts sequence. Moreover, it suffers most grievously by failing to make the course relevant in the way it could be, as I shall argue below. Closely related to this approach are the approaches of (1) exploiting the elegance and rigor of logic to make it an *aesthetics* course in the "beauty of mathematics" or else (2) an out-and-out high-powered course in the substance of mathematical logic. Against both it can be objected this is not the appropriate function of an introductory logic course in a philosophy liberal arts sequence. Against (1) it can also be said that few introductory logic instructors can successfully do it.

What I am suggesting is that it is a function of an elementary logic course to be relevant to the practical concerns of the average student, that the standard conception of logic courses intrinsically fails to be so relevant, and that none of the above alternatives does any better. If the relevance of introductory logic courses is to be teaching students to reason better, then I suggest we relegate our logic courses to the scrap heaps they belong in and replace them by "introduction to philosophy courses" which intensively teach one to do informal "philosophical analysis";[8] for if my argument above is sound, that is the essence of evaluating arguments, not formal logic.

3. Using Computers to Make Logic Relevant

An underlying bias in my preceding arguments is the idea that our elementary logic courses (as opposed to courses in mathematical logic required of majors[9] or advanced logic courses for those interested in mathematical logic) should enjoy a relevance appropriate to that of a humanities course satisfying a "liberal arts" requirement. What is the purpose of such liberal arts courses? This is a legitimate matter of debate, but I would suggest that a liberal arts course *minimally* should give the typical student some intellectual advantage in coping with the human environment he has to live in. The *promise* of the standard conception of logic courses clearly aims at satisfying this desideratum; unfortunately its *deliverance* does not.

Several years ago Professors Arthur Burks, Jaakko Hintikka, and I discussed this very issue: how do you make introductory logic courses relevant in the sense of the above conception of the purpose of a liberal arts sequence? Out of that conversation emerged a very interesting idea, which I wish to relate to you: At various times in human history there have been major technological innovations which have radically transformed the quality and nature of human existence. One need not go so far back as the Bronze Age for an example. In the Industrial Revolution of the 1760's onward, we find an example we can empathize with. But, to be more contemporary: since the ENIAC became operational in 1946, we have experienced an equally important example—the Computer Revolution. Today the computer is so persuasive in everyday life—the IRS audit of our taxes, the collation of credit and other personal information into computer files and other computer invasions of privacy, university records, industrial management decisions, bank statements, plotting war strategy, the manipulation of telephone switching, etc.—that in the last two decades the computer has radically changed the shape of human existence—especially in the technologically more advanced societies. Just as Henry Adams correctly found it necessary for the educated person to learn about, and come to grips with, the "Virgin and the Dynamo" if he was to adequately cope with his changed society,[10] so too the educated man today must have a clear understanding of computers—what their intrinsic limitations and capabilities are, and some clear idea how they do what they do.[11] For only with such understanding can we hope to cope with our computerized environment and realistically decide whether the likes of HAL of *2001* are more to be feared than our politicians.

How does one impart this important "humanistic" understanding of the non-human computer and its intrusions into human life? Learning to program a computer is little help—it only teaches you to talk with them, but gives you no insight into them except that they "speak" funny languages such as FORTRAN. And understanding the electrical engineering of micro-integrated circuits, thin-film or cryogenic memories, etc., is too much to expect of the average educated person. Out of that discussion with Burks and Hintikka emerged the following suggestion: There is a branch of logic, *automata theory* (the logic of computers) which—without getting involved in the engineering complexities of the actual construction of computers and without the ephemeral loss of touch with the actual internal operation of computers that comes from the study of programing languages—is ideal for efficiently imparting in fair logical detail a clear conceptual understanding of computers to the average college undergraduate: what their intrinsic capabilities and limitations are, and how they work. From such a study, a quite full conceptual understanding of computers can be gained. Granting that such an understanding of computers is of high humanistic priority, where is the best place to impart it in the college curriculum? Our answer was—in an elementary logic course. For on the basis of the minimal normal content of the typical elementary logic course, (propositional calculus, truth tables, disjunctive normal-forms, and a bit of naïve set theory—all of which I have found can be taught in 4-5 weeks), one can develop the substance of automata theory so as to impart such a conceptual understanding of computers.[12]

As an outgrowth of that germinal idea, the last several years I have been developing an experimental introductory logic course—first at the University of Illinois[13] and now at the University of Maryland—which minimally develops symbolic logic up through predicate calculus, then exploits the logic to develop automata theory—which in turn is used to impart a *conceptual* understanding of computers. The development of logic (including naïve set theory) and the automata theory are interspersed; and the automata theory developed includes the theories of switching nets, logical nets, Turing machines, Universal Turing machines, and a proof of the non-decidability of the halting problem. In the remaining time we consider modern prototypical computers; their esoteric applications in biological simulation, the computer compositions in music, artificial intelligence (including Samuel's Checker Playing Machine which learned to play tournament-caliber checkers); and finally the mind-machine problem ("Can Computers Think?"). In the course students learn the standard symbolic logic, though they spend less time on it and are much less proficient in using it to "evaluate" arguments in the usual question-begging ways. They do, however, come out knowing quite a bit about computers, can program the Universal Turing Machine, and have understood the philosophical significance of Church's Thesis and the non-decidability of the Halting Problem for Turing machines. And they have a healthy respect for computers, but have lost their awe of them. Most importantly, they react very favorably to the course, and feel they are getting something importantly relevant from the course (namely the understanding of computers), which is notably absent in their reactions to traditional logic courses. It is no wonder, then, that they are far more excited by and interested in this course than are students in traditional logic courses. And students who have studied computer programming since taking the course express the opinion that the course gave them an advantage over other students in learning the program. Thus student reactions vindicate this conception of using computers to make logic relevant.

4. Teaching A Computer-Related Logic Course

In arguing against the traditional conception of logic courses, and in pushing for the computer-related logic course, my motives obviously have not been merely to relate to you a novel and somewhat esoteric approach to teaching introductory logic. If my arguments and polemics against the standard conception of logic courses make their case, then either we must drop introductory logic from the curriculum (hopefully replacing it by courses in philosophical analysis), or else we must radically reconceive what we're trying to accomplish in such courses. And I have been urging a particular re-conception of introductory logic courses that I've found relevant in students' eyes and for which I think I can find reasonable academic justification. It should be obvious that my intention here is to convince the reader to attempt such a computer-oriented logic course, in the hopes that such courses eventually will become a standard part of the undergraduate philosophy curriculum.

There is, however, one practical problem facing the widespread incorporation of such a course into the undergraduate philosophy curriculum—namely that there does not presently exist a textbook suitable for the beginning undergraduate level course which comprehensibly develops the required automata theory out of the normal elementary symbolic logic. The problem is further complicated by the fact that existing automata theory textbooks[14] do not develop automata theory in a manner that easily can be adapted so as to be readily comprehended by typical undergraduates. Indeed, our experience in teaching the experimental course described above indicates that a nontraditional development of the material, unlike that found in existing automata textbooks, is a key element of its pedagogical success.[15]

Although this problem soon will be solved,[16] for the moment the instructor desiring to introduce the sort of computer-oriented logic course advocated here will have to teach it from duplicated notes he must prepare himself. By way of encouraging and aiding the introduction of such a course, I will briefly sketch the development of the course material which has proved successful in the experimental course outlined above.[17]

The logical apparatus required for developing the automata theory consists in rudiments of propositional calculus and naïve set theory. Any standard truth-table presentation of propositional calculus will suffice, though it is advisable to use '1' instead of 'T' and '0' instead of 'F'—as is standard in automata theory. In addition, the notion of a disjunctive normal form should be introduced, and an algorithm presented for obtaining disjunctive normal forms from truth tables.[18] The required set theory consists of the standard set operators, ordered *n*-tuples, and relations and functions.[19]

The first branch of automata theory to be developed is switching theory. This is done by introducing three basic switches—*not*, *or* and *and*. Their behaviors are defined by the truth tables for the corresponding sentential connectives, interpreting '1' as "on" and '0' as "off." Rules then are introduced for connecting these switches together to form switching nets.[20] The behavior of these switching nets can also be described by truth tables, which in turn allows wffs in propositional calculus to describe the behavior of these switching nets when net inputs are labeled by propositional variables. The truth-table algorithm for finding disjunctive normal forms (cf. above) is then adapted into a general procedure for constructing logical nets which realize net behaviors characterized by wffs of propositional calculus.[21]

Next the theory of switching nets is expanded into the theory of logical nets, which are constructed out of *and*, *or*, and *not* switches and a unit delay element which serves as a memory unit.[22] Then the Burks-Wang τ/λ techniques for describing the behavior of logical nets are introduced,[23] as are State diagrams.[24] Then Burks-Wang normal form logical nets[25] are introduced and exploited as a general procedure for constructing logical nets which realize behaviors specified via τ/λ techniques.

Turing machines are introduced by attaching tape read-write heads (having four inputs—"move right," "move left," "mark," "erase"—and one output which emits the contents of the cell under scan) to a logical net with one input and four outputs such that at most one output can ever be on.[26] It is observed, then, that all the logical net does in a Turing machine is to give the tape read-write head instructions to move its tape to the right or left, or mark a 1 or 0 on the tape, where the instruction given may depend upon the contents of the tapes. This means that the behavior of a Turing machine can be specified by a program of instructions to the tape read-write head. A variant of Wang's program-characterization of Turing machines[27] due to Thatcher[28] is then introduced, and subsequently used to work with Turing machines. Following Thatcher's development, it is shown how Turing machines can be used to compute functions. Then it is shown how to build a Universal Turing machine which, when suitably instructed, can compute any function which any Turing machine can compute.[29] Church's Thesis is then introduced and argued for,[30] from which discussion it is concluded that Turing machines can do anything any possible computer can do. Thus the capabilities of the Universal Turing Machine are exhaustive of the computational abilities of all possible computers. This establishes the intrinsic capabilities of computers. Next, the Halting Problem is introduced and show undecidable,[31] and it is shown how this establishes intrinsic limitations on what computers can do.

Having completed the automata theory, it next is shown how modern-day digital computers are realizations of automata,[32] and how the computational abilities of computers established in our study of Turing machines can be harnessed via numerical coding to perform a wide variety of practical tasks. For this part of the course I rely on the *Scientific American* reprints contained in Sections II and IV of Fenichel and Wizenbaum, *Computers and Computation*. This part of the course includes consideration of issues on artificial intelligence which, via a consideration of the parallels between logical nets and the human brain,[33] raises the mind-machine problem—Can machines think? Then the standard literature on this subject[34] can be considered; in dealing with this issue I find the approach taken by Arthur Burks in his APA Presidential Address[35] especially effective.

5. Summary

In this paper I have considered a number of standard conceptions of introductory logic courses and found them wanting. In Section 1, I argued the most common premise on which introductory logic courses rest—that formal logic provides an effective general means for evaluating arguments—was fraudulent, and that students recognize it as such. In Section 2, I considered various other frequently encountered conceptions of logic courses and found them unsatisfactory—at least if introductory logic courses are to function as part of required liberal arts general education sequences; for these approaches lack the relevance to the human situation essential to such courses. In Section 3, I suggested that there was a way to make introductory logic courses enjoy that relevance—namely by using the logic commonly taught in such courses to develop automata theory, and then use that automata theory to impart a thorough conceptual understanding of computers. Section 4 discussed the feasibility of such a course, pointing out a practical problem caused by the temporary unavailability of an adequate textbook for such a course—which requires that it be taught from notes prepared by the instructor. A somewhat detailed outline of such a course was presented, indicating the sources an instructor should consult in working-up such a set of notes. My experience has shown the course can be taught quite successfully from such notes distributed to the class. Working-up such a set of notes requires effort, but expending the effort is morally preferable to the easier practice of continuing to serve our students the academic fraud kown as the standard elementary logic course.

Notes

This is a revised version of part of an invited address given before the Western Conference on the Teaching of Philosophy in conjunction with the 1973 Western Division meetings of the American Philosophical Association. I am grateful to my former colleague at the University of Illinois, Professor Hugh Petrie, for comments on a draft of this paper. This paper is dedicated to Stephen Toulmin—who although trained as a mathematician, has developed a healthy disrespect for mathematical logic as a philosophical tool.

1. D. Kalish and R. Montague, *Logic: Techniques of Formal Reasoning* (New York: Harcourt, Brace, and World, 1964), 3. The spirit in which this passage obviously is intended in the overall context of the book should not be overlooked.

2. Copi, *Symbolic Logic*, 3rd edition (New York: Macmillan 1967), 1–2; emphasis added.

3. Kalish and Montague, *Logic*, 176.

4. *The Advisor* (Urbana: University of Illinois Press, 1969–1970), 239.

5. See especially chap. 5 of his *Elementary Logic* (New York: Oxford, 1965).

6. This argument is from B. Mates, “Synonymity,” reprinted at L. Linsky, *Semantics and the Philosophy of Language* (Urbana: University of Illinois Press, 1952), 111–136; the discussion which follows is based on Section III of that paper.

7. See P. F. Strawson, *Introduction to Logical Theory* (London: Methuen, 1952).

8. From my own experience, such courses do a better job of improving reasoning ability than standard logic courses.

9. I’m not sure there is any rationale for requiring courses in mathematical logic of them, except the fact it is a kind of literacy now required to read portions of the philosophical literature. And for this a good twenty-five page set of mimeographed notes on logic is enough. In this vein, while I was teaching at the University of Illinois (Urbana) we revised our Ph.D. logic requirement there. After discussion, the committee drafting the new requirement agreed the ideal logic requirement for philosophers *qua* philosophers was to prohibit any knowledge of symbolic logic at all. Feeling this was unenforceable and unrealistic, it finally was decided that we would require at least as much logic as was contained in Vol. 1 of A. Church, *Introduction to Mathematical Logic* (Princeton: Princeton University Press, 1956). For in that case we could rightly hold them responsible for any misuse of logic they might make!

10. See his *The Education of Henry Adams* (New York: Modern Library, 1931).

11. The importance of this need is reflected by the fact that the Smithsonian Institution’s Museum of Science and Technology devotes as much display space to computers

as it does to medicine and pharmacy, to petroleum, or to the origins of modern science. The emphasis in their displays is on understanding how computers work.

12. This is no accident; historically, the first computers were, from an engineering point-of-view, simply devices to do automata theory; and the first computers were designed by teams of logicians and engineers building machines to do logic. Since then, automata theory has emerged as an abstract field of computer design, though it has tended to supplant logic by its algebraic extensions in the field of automata theory.

13. In developing the course and trying it out on students at the University of Illinois, I was aided by Professor Thomas Nickles and my graduate assistants Mr. Michael Drummer and Mr. Richard Schubert.

14. For example, M. Davis, *Computability and Undecidability* (New York: McGraw-Hill, 1950), or M. Minsky, *Computation: Finite and Infinite Machines* (Englewood Cliffs: Prentice Hall, 1967).

15. The situation is not unlike that encountered in introducing symbolic logic into the undergraduate curriculum. So long as textbooks only developed symbolic logic axiomatically along the line of *Principia Mathematica*, there were extreme practical difficulties facing the introduction of symbolic logic into the undergraduate curriculum. However, once textbooks were introduced which presented non-standard developments of symbolic logic along Gentzen natural-deduction lines, symbolic logic courses became standard undergraduate offerings.

16. Namely by the textbook for such a course I am writing, *Introduction to the Logic of Computers*.

17. The version of this paper read before the Western Conference on the Teaching of Philosophy (cf. unnumbered note above) was approximately eight times longer than the present version, and contained a quite detailed development of the course material sketched below. Copies of the longer development can be obtained by writing the author.

18. Such an algorithm is presented at G. Massey, *Understanding Symbolic Logic* (New York: Harper and Row, 1970), 43–46.

19. A sufficient development can be found in R. Stoll, *Sets, Logic, and Axiomatic Theories* (San Francisco: Freeman, 1961), Sections 1.1–1.6, 1.8.

20. The rules are rules 1, 2, 4, and 5 of the definition of a logical net given in Section 4. of I. Copi, C. Elgot, and J. Wright, "Realization Events by Logical Nets," in E. F. Moore, *Sequential Machines* (Reading: Addison-Wesley, 1964), 175–192.

21. Various aspects of this development of the theory of switching nets can be found in Copi, Elgot, and Wright, "Realization Events by Logical Nets,"and the works cited in Note 23 below, as well as in the longer version of this paper cited in note 17.

22. Cf. Copi, Elgot, and Wright, "Realization Events by Logical Nets," for the definition.

23. Cf. Section 2.2. of A. Burks and H. Wang, "The Logic of Automata," in H. Wang, *A Survey of Mathematical Logic* (Amsterdam: North Holland, 1962), 175–223.

24. Cf. Section 1. of R. McNaughton and H. Yamada, "Regular Expressions and State Graphs for Automata," Moore, *Sequential Machines*, 157–174.

25. Cf. Burks and Wang, "The Logic of Automata," Section 2.3.

26. For an intuitive description of Turing machines, see Wang's article in R. Fenichel and J. Weizenbaum, *Computers and Computation* (San Francisco: Freeman, 1971), 136–144 and Section 6.0 of Minsky, *Computation*.

27. Cf. H. Wang, "A Variant to Turing's Theory of Computing Machines," *Journal of the Association for Computing Machinery* 4 (1957): 63–92.

28. Cf. Section 1 of J. Thatcher, "Self-Describing Turing Machines and Self-Reproducing Cellular Automata," in A. W. Burks, *Essays on Cellular Automata* (Urbana: University of Illinois Press, 1970), 103–131.

29. This is done along the lines of Thatcher, ibid., Section 6, except that the Universal Turing Machine presented operated on a three-digit binary code. See the longer version of this paper noted in Note 17 for the details of this design.

30. Cf. A. Fraenkel and Y. Bar-Hillel, *Foundations of Set Theory* (Amsterdam: North Holland, 1958), 297–303; chaps 11–13 of S. Kleene, *Metamathematics* (Princeton: Van Nostrand, 1950); and chap. 5 of Minsky, *Computation*, for Church's Thesis (also known as Turing's Thesis) and the evidence in support of it.

31. The proof follows Thatcher, "Self-Describing Turing Machines," Section 4.

32. Cf. Section 1 of Fenichel and Weisenbaum, *Computers and Computation*.

33. Cf. W. McCulloch and W. Pitts, "A Logical Calculus of Ideas Immanent in Nervous Activity," *Bulletin of Mathematical Biophysics* 5 (1943): 115–133; and John von Neumann, *Computer and the Brain* (New Haven: Yale University Press, 1958) and *Theory of Self-Reproducing Automata* (Urbana: University of Illinois Press, 1966), Part 1.

34. Cf. A. R. Anderson, *Minds and Machines* (Englewood Cliffs: Prentice Hall, 1964).

35. Cf. his "Logic, Computers, and Men," in the *Proceedings and Addresses of the American Philosophical Association* 46 (1972–1973): 39–57.