CANTOR'S DIAGONAL ARGUMENT: PROOF AND PARADOX

Cantor's diagonal method is elegant, powerful, and simple. It has been the source of fundamental and fruitful theorems as well as devastating, and ultimately, fruitful paradoxes. These proofs and paradoxes are almost always presented using an indirect argument. They can be presented directly. The direct approach, I believe, (1) is easier to understand, (2) unifies the proofs and paradoxes by exhibiting a single strategy, (3) shows Russell's paradox to be the (obvious?) ultimate, set theoretic application of the method, and (4) is extendable to some of the semantic paradoxes.

For a first example consider all numbers between zero and one which can be represented by terminating decimals, e.g. 0.25, or infinite, repeating decimals, e.g. 0.123123123. . . (you may or may not know the name of such numbers). It is convenient to use infinite representations in all cases, so we append an infinite number of zeros at the end of each terminating representation, making all of our decimals infinite, repeating. Now suppose we list all such representations, i.e. all infinite, repeating decimals, with integer part zero. It may not be obvious how to do this, but it is doable, in many different ways, one of which is

0.**0**000000000000. . .
0.9**9**99999999999. . .
0.50**0**0000000000. . .
0.333**3**333333333. . .
0.6666**6**66666666. . .
0.25000**0**0000000. . .
0.750000**0**000000. . .
0.2000000**0**00000. . .
0.40000000**0**0000. . .
0.600000000**0**000. . .
0.8000000000**0**00. . .
0.16666666666**6**6. . .
0.833333333333**3**. . .
.
.
.

(The bold digits are explained below. The method used above to list all the decimals may not be obvious, but its discovery is, as they say, left to the reader.)

We now use Cantor's method to diagonalize out of this list by constructing, digit by digit, a decimal representation that is not on the list: if the i-th decimal digit in the i-th decimal representation is a 3, make the i-th digit in the construction a 7; and if the i-th decimal digit in the i-th decimal representation is not a 3, make it a 3 (giving the method the name 'diagonalization'). This produces

0.3337333333337. . .

As you might know all of the decimals in the list represent rational numbers. The constructed decimal is clearly not on the list, differing from the i-th decimal at least in the i-th digit. Does it represent a rational number? As you might also know the decimals on the list represent *all* rational numbers (between zero and one), so the constructed decimal can not represent a rational.

A set which can be put into an infinite list is called *countably infinite*. Thus, the rationals are countably infinite. (We have only discussed numbers between zero and one, but decimal representation of all rationals can also be listed.)

In 1874 Cantor used his method to prove the very significant result that the reals are *not* countable. What is needed is an argument that *no* infinite list of decimals can contain representations of all reals. The stagey is to show that any such list must be incomplete. But this is obvious, since any list can be diagonalized out of; i.e., given an arbitrary list of decimals (for simplicity, again, consider numbers between zero and one), the diagonally constructed decimal is not on the list, and therefore, represents a real not on the list:

Theorem 1 – Cantor (1874). *The set of reals is uncountable.*

The diagonal method can be viewed in the following way. Let ***P*** be a property, and let *S* be a collection of objects with property ***P***, perhaps all such objects, perhaps not. Additionally, let *U* be the set of *all* objects with property ***P***. Cantor's method is to use *S* to systematically construct an object *c* which is not in *S*, called *diagonalizing out of the set*. In our first example ***P*** is the property rational number, *U* is the set of all rational numbers, and *S* is *U*. The conclusion we're forced to here is that c must not be in *U*.

In the second example ***P*** is the property real number, *U* is the set of all reals, and *S* is an arbitrary infinite list of elements from *U*. Since *c* is always in *U*, our conclusion is that *S* can never be complete.

These are the two uses of Cantor's method for proving theorems. For a frivolous, but instructive, example consider the one given by Douglas Hofstadter in *Gödel, Escher, Bach*, p. 404. Let ***P*** be the property 'famous mathematician', and let S consist of the following people who have property ***P***:

De Morgan
A**b**el
Bo**o**le
Bro**u**wer
Sier**p**inski
Weier**s**trass

Now diagonalize out of this list by moving back one letter in the alphabet. If you believe the above list is complete, that it contains all famous mathematicians, you must conclude Cantor is not a famous mathematician. If on the other hand you believe Cantor *is* a famous mathematician, you must conclude the list is not complete.

To review, we have the two possibilities: (1) if S is U, then c can't have ***P***, or, contrapositively, (2) if the constructed object *c* has property ***P***, then *S* couldn't have been all of *U*. As mentioned these two cases form the basis of the proofs using diagonalization. A third possibility, that *c* has property ***P***, and *S* isn't all of U, is uninteresting. A fourth case is an impossible one: *c* has ***P*** *and S* is *U*. This is paradox. Looked at another way, if we diagonalize out of the *universe*, we're in trouble.

Before moving to paradox let's look at one more theorem. The motivation for the name 'diagonal method' is obvious from the above examples, however, as mentioned, the essence of the method is the strategy of constructing an object which differs from each element of some given set of objects. We now employ the diagonal method to prove Cantor's arguably most significant theorem:

Theorem 2 – Cantor's Theorem (1891). *The power set of a set is always of greater cardinality than the set itself.*

Proof: We show that no function from an arbitrary set S to its power set, $\wp(U)$, has a range that is all of $\wp(U)$. nThat is, no such function can be *onto*, and, hernce, a set and its power set can never have the same cardinality.

To that end let f be any function from S to $\wp(U)$. We now diagonalize out of the range of f. Construct a subset C of S, i.e. an element of $\wp(U)$, which is not in the range of f as follows: for any $a \in S$ make C differ from $f(a)$ with respect to the element a itself. I.e. if $a \in f(a)$, then don't put it in C; however, if $a \notin f(a)$, put it in C. Symbolically,

$$C = \{a:\ a \in S \text{ and } a \notin f(a)\}.$$

Clearly C differs from each element in the range of f (with respect to at least one element). Since f is arbitrary, we conclude there can be no function from S onto $\wp(U)$. Thus, every set has cardinality smaller than its power set.

This theorem was proved by Cantor in 1891 and led him in 1895 to what is now known as Cantor's paradox: if U is the universe of all sets, then every subset of U, being a set, is an element of $\wp(U)$. Thus, $\wp(U) \subset U$, so (it would seem) the cardinality of $\wp(U)$ cannot be greater than the cardinality of U. But this contradicts Cantor's theorem. (For an interesting discussion of why Cantor did not see this and other set-theoretic paradoxes as negative (until Russell's), but as positive results, see Dauben [1, pp. 242-247].) This paradox was not published until 1932, but word of its discovery spread and reached Russell in 1901, whereupon he constructed his paradox. Russell's paradox can be seen as the ultimate, set-theoretic application of Cantor's diagonal method: diagonalize out of the *Universe*!

Russell's Paradox (1901). Construct a set which differs from *every* set, including itself! Let R be the set to be constructed. Now, for any set S, how shall we make R differ from S? With respect to what can we make R differ from S? How about S itself? If $S \in S$, we won't put S in R. On the other hand, if $S \notin S$, we will put S in R. Symbolically,

$$R = \{S:\ S \notin S\}.$$

Now R differs from each set (with respect to at least one element), including itself – paradox! (How does R differ from R? Well, if $R \in R$ we didn't put R in R, and if $R \notin R$ we put R in R. Either way, R can't equal R .)

Russell [4, p. 362] states he was led to his paradox by applying Cantor's proof to the universe of sets. Perhaps he reasoned as above, or alternatively, as follows: There must be a mapping from U *onto* $\wp(U)$ – the *identity* map is one such (since every subset of U is a set.) Russell's paradoxical set is then generated by applying the construction in Cantor's theorem to this map.

The argument Cantor had used to produce an elegant and powerful theorem was modified slightly and used by Russell to produce an elegant and devastating paradox. Cantor's theorem had guaranteed an unlimited collection of sizes of infinity, giving good evidence that this new theory would be rich and fruitful. Russell's paradox, because of its crushing blow to the new field and its intimate connection with efforts to shore up the foundations of mathematics, was a major force behind the flurry of activity in the foundations of mathematics in the first half of the twentieth century. Much of the effort was directed toward rescuing set theory. We quote Hilbert [2, p. 141]:

"Whenever there is any hope of salvage, we will carefully investigate fruitful definitions and deductive methods. We will nurse these, strengthen them, and make them useful. No one shall drive us out of the paradise which Cantor has created for us."

Russell's paradox leads to the following argument. Given any set S, we can diagonalize out of it. Construct an object which is not in S by creating a set which is different from each element of S. Of course if the elements of S are not sets, then any set will be different from these elements; but we can describe a method which will work regardless of the character of the elements of S. Call the set to be constructed R_S. For each element s of S, determine whether or not $s \in s$. This will of course be false for any element which is not a set, as well as for many sets. Now, we put s in R_S iff $s \notin s$. Symbolically,

$$R_S = \{s: \ s \in S \text{ and } s \notin s\}.$$

Then R_S differs from each element of S. Since many sets (all? – see below) do not contain elements (sets) which are self-containing, R_S is often (always?) S itself. (In most cases the diagonal method is used in a context requiring the diagonal element to be similar to the elements of the given set, in addition to being distinct from each of them, as in the first two theorems. Here, however, the only requirement is to get out of the set.) Thus, we can diagonalize out of any set; i.e. given any set, there is an object (set) which is not an element of that set. In particular there can be no set which contains every set – any set can be diagonalized out of. This is just Russell's paradox. We can therefore consider Russell's paradox (as well as Cantor's) as a *reductio ad absurdum* proof of

Theorem 3. *There is no universal set.*

Since the specification

$$\{x: \ x \text{ is a set and } x = x\}$$

defines the contradictory universal set, such specifications had to be prohibited in the axiomatic reconstruction of set theory. Furthermore, even if the universal set is disallowed, Russell's argument still presents paradox, still diagonalizes out of the *universe* (if not out of the universal set). The solution for most axiomatic set theories is, again, to prohibit the specification of Russell's paradoxical set. In fact these theories usually prohibit sets which contain themselves. (Russell's set is self-containing iff it isn't!) While an infinite set can be thought of as having "infinite breadth" (if all its elements could be named, the roster name of the set would be infinitely long), a self-containing set would have "infinite depth" (if $S \in S$, then S would have infinite nesting – $S \in S \in S \in \ldots$), as does a picture which depicts itself. One might argue that while such pictures can be conceived of, they cannot be "realized." Similarly, most mathematicians in foundations do not believe such sets exist. Consequently they prohibit self-containing sets. Actually they prohibit, in general, infinite, descending $\in$ – chains (no set has infinite depth).

It might be helpful to see a non-mathematical analogy. Russell produced an interesting variation of his paradox whose subject matter is not mathematics. Consider properties (predicates), e.g. red, long, controversial, abstract, etc. Some properties apply to physical objects (e.g. red and long), while others apply to abstractions such as ideas, concepts, or properties (e.g. controversial or abstract). We want to diagonalize out of the collection of *all* properties, giving paradox.

To that end define a new property, *impredicable*, meant to apply not to physical object but to the abstractions, properties. And, of course, we want to define it in such a way that it is different from *every* property. Just as we define the property *red* for a child by telling him or her which physical objects are and which are not red, we define the property *impredicable* by saying which properties are and which are not impredicable.

Thus, given a particular property such as *controversial*, we make (the property) impredicable differ from (the property) controversial with respect to the property of being controversial. Since there is nothing controversial about the property controversial, we include controversial as one of those properties which has the property of being impredicable; i.e. controversial is impredicable. (I think you'll also agree, before we finish, that impredicable is controversial.) Hence, the two properties, impredicable and controversial, differ in (at least) the following way: controversial has the first of these two properties but lacks the second.

For a second example consider the property *abstract*. Since (the property) abstract it is itself abstract (*all* properties are abstract), we will say that it is *not* impredicable. Hence, impredicable differs from abstract in that (the property) abstract is abstract, but abstract isn't impredicable.

In general, then, a property is impredicable iff it isn't self-descriptive. Thus, impredicable differs from *every* property, including itself – paradox! (It differs from itself in that it is impredicable iff it is *not* impredicable.)

We can symbolize the definition of impredicable if we consider properties *extensionally*, i.e. if we identify the property with the collection of objects which have the given property. Thus, we will identify the property impredicable with a certain set, viz. the set of all objects (properties) which are impredicable. Then the statement 'controversial is impredicable' can be translated as

$$\text{controversial} \in \text{impredicable},$$

and 'abstract is not impredicable' is translated as

$$\text{abstract} \notin \text{impredicable}.$$

Now, letting 'V' denote the collection of all properties, impredicable can be defined as follows:

$$\text{impredicable} = \{P : \ P \in V \text{ and } P \notin P\}.$$

The analogy with Russell's set-theoretic paradox is almost exact.

In 1908 Grelling published a very similar paradox which is concerned with adjectives (words) rather than properties. Define the adjective 'heterological' to apply to an adjective iff the adjective does not apply to itself. This is very similar to Russell's definition, but the "extension" of the adjective 'heterological' is very different from the "extension" of the property impredicable. For one thing their extensions contain different kinds of objects, words in one case and concepts in the other. But there is a more subtle difference. For example, the *adjective* 'polysyllabic' is not heterological (since the word 'polysyllabic' is polysyllabic). But the *property* polysyllabic *is* impredicable (since the concept of being polysyllabic is not itself polysyllabic – concepts don't have syllables). And this difference occurs precisely because the one applies to words and the other to concepts.

As was mentioned earlier, Russell's variant on his paradox is not mathematical in nature; the same is true of Grelling's. They are among the so-called *semantic* paradoxes. These semantic paradoxes did not have the direct impact on the foundations of mathematics of the set-theoretic paradoxes; however, they did play a crucial role in the subsequent analysis and development of foundations. Fraenkel and Bar-Hillel [3, p. 12] state, ". . . in one of the most interesting developments in modern foundational research it became clear that the problem presented by the semantic antinomies (paradoxes) . . . served as the starting point for investigations of immense direct impact on modern mathematics." For general discussions of both semantic and set-theoretic paradoxes, see [3, pp. 1-18] and [1, pp. 481-518].

Let us leave paradox now and return to theorem. A slight modification of the proof of the uncountability of the reals shows that the set of (single variable) number-theoretic functions is not countable.

Theorem 4. *The set* $F = \{f : f\colon \mathbf{N} \to \mathbf{N}\}$ *is uncountable.*

Proof: As before we show that any list $f_1, f_2, f_3, \ldots$ of elements of F is not complete (there is no function from **N** to F which is *onto*). Think of a vertical list of functions, with the values of each function enumerated horizontally, as a doubly-infinite array (like a list of infinite decimals). Now construct a function which is not on the list by changing the diagonal: let

$$f(i) = f_i(i) + 1 .$$

This function clearly differs from each function on the given list (for at lest one argument value). Thus, no list can be complete, and therefore, F is uncountable. (In fact F is the same size as the reals.)

Starting with a close analog of the last theorem, we now consider a number of theorems concerning *computable functions* and *computable lists*. I will use the term *computable* in an informal way and assume that associated with each computable function, or list, is an *algorithm* (also taken intuitively) which computes the function or generates the list. Intuitively, an algorithm is a list of instructions which specifies (unambiguously) how to perform some task, e.g. finding the values of a particular function or generating a particular list. We assume our algorithms are written in a particular language such as the informal language within which we do mathematics (English, or another natural language, supplemented by technical terms) or a formal language such as a computer language. We also assume an algorithm can contain only a finite number of instructions. Thus, an algorithm is a finite, syntactic object: a finite string of symbols from some (finite) alphabet. Of course many algorithms may compute the same function or generate the same list.

The formal counterparts in recursion theory, and sometimes automata theory, will be indicated parenthetically. That the set of recursive functions, the set of Turing-computable functions, and the set of functions calculable (in an ideal sense) by any general-purpose programing language are all the same set is a theorem of mathematics. That this common set is precisely the set of *computable* functions is the universally accepted hypothesis (definition) known as the Church-Turing thesis.

Since an algorithm is a finite string of characters from a finite alphabet, the number of computable functions is countable. It would seem that we can diagonalize out of this universe, while at the same time remaining within it! This was a serious problem in the 1930's. Quoting Hartley Rogers from his classic book on recursion theory [5, p.11]:

> It is evident that the diagonalization method has wide scope, for it is applicable to any case where the sets of instructions . . . can be effectively (i.e., algorithmically) listed. At first glance, it is difficult to see how a formal characterization can avoid such effective listing and still be useful. The diagonal method would appear to throw our whole search for a formal characterization into doubt. It suggests the possibility that no single formally characterizable class of algorithmic functions can correspond

exactly to the informal notion of algorithmic function....
These are some of the considerations and difficulties, albeit vague and informal, that surround the problem of getting a satisfactory characterization of algorithm and algorithmic function. They had to be faced by the mathematicians who first addressed themselves to that problem in the 1930's, mathematicians who were stimulated in their work by recent successes of formal logic and its methods.

Let us follow the resolution of the problem. The first result concerns a subset of the computable functions, viz. those which are computable using only *bounded* loops (the primitive recursive functions). In a bounded-loop algorithm any loop must have a predetermined upper bound (determined before the loop is entered) on the number of times the loop will be executed. As an example of an algorithm with an unbounded loop consider the following procedure to find the next prime after *n*: starting with *n*+1, test each successive integer for primeness until one is found. Of course there is a bounded-loop algorithm which also calculates this function, since *n*!+1 can be used as a bound. Note that bounded-loop algorithms must always halt. One *free-loop* algorithm which may or may not always halt is the one which searches for the next twin-prime pair.

As mentioned earlier, the entire set of computable functions is countable. Hence we won't be proving that any subset of computable functions is uncountable. Nevertheless, Cantor's method is extremely useful, producing arguments which seem at first, like Russell's, to be headed toward disaster. We quote Rogers again, from page 31: "It is not inaccurate to say that our theory is, in large part, a 'theory of diagonalization'."

Now consider a listing $f_1, f_2, f_3, \ldots$ of bounded-loop computable (primitive recursive) functions. The diagonal function

$$f(i) = f_i(i) + 1$$

is clearly not on the list, and hence, is not bounded-loop computable. But is it (intuitively) computable? It is if we can generate, in a computable way, the list $f_1, f_2, f_3, \ldots$ (if the set is recursively enumerable): to compute $f(n)$ merely generate f_n, compute $f_n(n)$, and add one. To generate such a list, start listing the finite sequences of characters of the language in which the algorithms are written, by length, and alphabetically within each length. As this is being done, throw out those which are not valid algorithms (of one numeric parameter and giving one numeric output) and those valid algorithms which have unbounded (free) loops. We assume this can be done. If the Church-Turing thesis is granted, the algorithms can be thought of as, say FORTRAN programs. And to make loops easier to check for boundedness, we can eliminate form the programming language all branching except that connected with DO-loops. A program similar to a compiler could carry out these processes (the primitive recursive functions are recursively enumerable). Each algorithm (considered as a specific sequence of characters) appears only once on the list but each function appears many times, i.e. the list $f_1, f_2, f_3, \ldots$ contains many duplicates. But the point is we have listed the bounded-loop computable We have therefore proved

Theorem 5. *There are computable (recursive) functions which are not bounded-loop computable (primitive recursive).*

Thus, we are able to diagonalize out of the set of bounded-loop computable functions while remaining within the domain of computable functions. But why stop here? Maybe we can diagonalize out of the entire set of computable functions, while at the same time remaining with that set, producing paradox!

To that end, assume $g_1, g_2, g_3, \ldots$ is a listing of *all* computable functions. Then the diagonal function

$$g(i) = g_i(i) + 1$$

is clearly not on the list. This is the concern expressed by Rogers above. But to produce paradox, to argue we have found a computable function which is not on the list of *all* computable functions, we must have that $g_1, g_2, g_3, \ldots$ is computably listable. Thus to avoid paradox we must conclude

Theorem 6. *The set of algorithms which are defined for all (halting Turing machines, recursive functions) is not computably listable (r.e.).*

It could be objected that the above argument does not justify the conclusion of the theorem; that it is equally valid to conclude that the notion of *computable* is flawed in some way, similar to the conclusion derived from Russell's paradox about Cantor's naive conception of *set*. However, if the theorem is stated in terms of recursive functions or Turing machines, and the Church-Turing thesis is accepted (implying that intuitively computable functions are recursive or Turing-computable), the conclusion is justified (see below). Notice we can also conclude from the above argument that there are functions which are not computable; *g* is one such. Theorem 4 already tells us that "most" functions are not computable, but now we have a specific example.

From Theorem 6 a further conclusion can be drawn: when generating a list of all algorithms, we cannot (in a computable way) eliminate those that fail to halt (or produce invalid output) for some input values. (As any beginning programing student knows, it is not easy to tell if a program contains infinite loops.) This is a statement of the recursive unsolvability of one form of the famous Halting Problem:

Theorem 7. *There is no computable procedure which can determine, for each algorithm, whether or not it will halt (with a valid output) for all input values.*

This then is the solution to the problem mentioned by Rogers. The concept "algorithm" can be defined in such a way that deciding whether a given (finite) syntactic object is or is not an algorithm is a computable (algorithmic) procedure. But this can be done only by including non-terminating algorithms, i.e., algorithms with calculate *partial* functions. And then we cannot sort out, in a computable manner, the terminating algorithms (total functions) from the non-terminating ones (partial functions).

Since the partial functions can be computably listed, let us try to diagonalize out of this set. Let $h_1, h_2, h_3, \ldots$ be a computable listing of all partial, computable functions. Our first candidate for an "escape" function, $h(n) = h_n(n) + 1$, fails to be well-defined since $h_i(i)$ may not be defined for some i. The attempted fix,

$$h(n) = \begin{cases} h_n(n) + 1, & \text{if } h_n(n) \text{ is defined} \\ 0, & \text{otherwise} \end{cases}$$

does produce a well-defined, partial function which is different from every computable, partial function. However, we again avoid paradox since *h* is not computable, this time because the Diagonal Halting Problem is recursively insolvable:

Theorem 8. *Given a computable list of algorithms, there is no algorithm which will tell if the i^{th} algorithm will halt for input i.*

(As in the argument for Theorem 6, we have produced a function, in this case partial, which is not computable.)

Thus we have the well-defined set of total, computable functions for which Cantor's argument does not produce paradox – the diagonal function is not computable since the set is not computably listable (r.e.); and the well-defined, computably listable (r.e.) set of partial, computable functions which also does not produce paradox – the diagonal function is again not computable since we can't always decide, algorithmically, whether an algorithm will stop (the Diagonal Halting Problem is recursively unsolvable).

Summary

In Theorems 1 and 4 we considered arbitrary lists of particular sets of objects. In each case we found an object in the set not on the list. We thus concluded that no list of elements of the set can be complete. Cantor's theorem is similar but applies to *any* set (not just ones that can be listed), guaranteeing no largest cardinal. And if the *universe* of sets is used, leaving us nowhere to go when we diagonalize, Russell's paradox is produced. Russell's argument stimulated the development of axiomatic set theory, as well as foundational work in general. His argument can be interpreted as a *reduction ad absurdum* proof of the nonexistence of his specific set, or (modified slightly) a direct proof that there is no universal set. The modern generalization of this result is that there are no sets of "infinite depth" – no infinitely descending epsilon chains.

Theorem 5 is similar to Theorem 4, but the switch to *computable* functions produces a significant difference. There is no question about the countability (listability) of the given set. But here, in order to stay within the given domain (computable functions) when diagonalizing, the *list* must be computable (the set must be r.e.). That is, the diagonal function is guaranteed to be computable only if the list is computable. In Theorem 5 the list (of bounded-loop computable functions) *is* computable, and therefore, we successfully diagonalize out of the set while remaining in the desired domain of computable functions, giving that there are computable functions which are not bounded-loop computable.

In Theorem 6 by trying to diagonalize, in a computable way, out of the set of *all* (total) computable functions (shades of Russell), we produce the initially surprising result (referred to by Rogers) that the set is not computably listed. This argument also gives the unsolvability of one form of the Halting Problem. In Theorem 8 we move to partial functions. This list is computable, so the conclusion to the diagonal argument must change once again. Since the functions are partial, the standard diagonal function is not well-defined. And when we fix it up so that it is well-defined, we introduce another problem whose recursive solvability is open to question. We therefore conclude that the Diagonal Halting Problem is unsolvable.

As we have seen, the diagonal method created by Cantor has been tremendously fruitful, not only in its original context of set theory, but throughout foundations, including its development to create an entire new field, the important theoretical and applied subject of computability/recursion theory. (The method also plays a crucial role in Gödel's fundamental incompleteness theorems.) Hilbert [4, p. 139] recognized Cantor's unique talents: "This theory (set theory) is, I think, the finest product of mathematical genius and one of the supreme achievements of purely intellectual human activity."

References

1. E.W. Beth, *The Foundations of Mathematics,* North-Holland Amsterdam, 2nd ed., 1968.

2. Joseph Warren Dauben, *Georg Cantor: His Mathematics and Philosophy of the Infinite,* Harvard University Press, 1979.

3. Abraham A. Fraenkel and Yehoshua Bar-Hillel, *Foundations of Set Theory,* North Holland, Amsterdam, 1958.

4. David Hilbert, "On the Infinite," in *Philosophy of Mathematics,* ed. by P. Benacerraf and H. Putnam, Prentice-Hall, Englewood Cliffs, N.J., 1964.

5. Douglas Hofstadter, *Gödel, Escher, Bach*, Basic Books, 20th Anv. Edition, 1999.

6. Harley Rogers, Jr., *Theory of Recursive Functions and Effective Computability,* McGraw-Hill, New York, 1967.

7. Bertrand Russell, *Principles of Mathematics,* Norton, New York, 1937 (first published in 1903).

References

1. Joseph Warren Dauben, *Georg Cantor: His Mathematics and Philosophy of the Infinite,* Harvard University Press, 1979.

2. David Hilbert, "On the Infinite," in *Philosophy of Mathematics,* ed. by P. Benacerraf and H. Putnam, Prentice-Hall, Englewood Cliffs, N.J., 1964.

3. Douglas Hofstadter, *Gödel, Escher, Bach*, Basic Books, 1979.

4. Bertrand Russell, *Principles of Mathematics,* Norton, New York, 1937 (first published in 1903).

(The following are included in my other version but don't seem to be used here:

E.W. Beth, *The Foundations of Mathematics,* North-Holland Amsterdam, 2nd ed., 1968.

Abraham A. Fraenkel and Yehoshua Bar-Hillel, *Foundations of Set Theory,* North Holland, Amsterdam, 1958.

Harley Rogers, Jr., *Theory of Recursive Functions and Effective Computability,* McGraw-Hill, New York, 1967.)