Here you will find the lecture by Professor SquarePunkt, followed by a discussion of its contents.
L.K.

##

Cantor's Demise

by Professor Hilbert Squarepunkt

We have seen how Cantor's diagonal argument can be used to produce new elements that are not on a listing of elements of a certain type. For example there is no complete list of all Left-Right sequences of the form A1A2A3... where An = L or R and two such sequences A and B are said to be equal when An = Bn for all n=1,2,.... We proved this by assuming that we had a list of such sequences A(1),A(2),... such that A(n)m denotes the m-th element of the sequence A(n). Then we constructed the *diagonal sequence* D defined by Dn = A(n)n. And we made the *flipped diagonal sequence* Flip(D) from this by defining Flip(D)n = L when Dn = R and Flip(D)n = R when Dn = L. Cantor argues that Flip(D) is necessarily a new sequence not equal to any Dn that is on our list. The proof is clear, since Flip(D) is constructed to differ from each sequence in the list in at least the n-th place for D(n).

Now Cantor's intent was to prove that the real numbers are uncountable (not listable) and I have discovered a fatal flaw in his argument! Let me explain. I will use the binary notation for real numbers between 0 and 1. Thus
.0= 0
.1 = 1/2
.01 = 1/4
.001 = 1/8
.0001 = 1/16 and generally
.0000...01 = $1/2^n$ where there are n-1 zeros before the 1.
Then real numbers between zero and one are represented by binary analogues of decimals like .101001000100001...
Note that .0111111... = .1000....
since 1/4 + 1/8 + 1/16 + ... = 1/2.

We apply the Cantor argument to lists of binary numbers in the same way as for L and R. In fact L and R are analogous to 0 and 1. For example if we had the list
(1) .10111...

(2) .10101010...
(3) 0.1110110011...
...
Then we would make the diagonal sequence
D = .101 ...
and flip it to form
Flip(D) = 0.010...
Just as we argued before Flip(D) is not on the list, and so
the list is incomplete.

Here is a counterexample to Cantor's argument!
Consider the following list:
(1) .10000000....
(2) .00100...
(3) .0001000...
(4) .00001000...
(5) .000001000...
...
As you can see this is a very definite list. The first element of the list is .10000... and subsequent members of the list consist in n-zeros, a 1 and then zeros forever. Ok? Now the diagonal element is
D = **.1000...** and
Flip(D) = **.01111...**
But **.0111... = .10000**
and so Flip(D) = D = **.1000...** and so
Flip(D) = the first element of our list!
Flip(D) is not a new number outside he list.
This is a counterexample to Cantor!

Gentleman, Ladies: Cantor lies in ruin before us. His theory is a shambles. No longer can we speak of higher infinities. Mathematics must be revised from the ground up and built anew. I exhort you to join me in this grand project of reconstruction of the truth.
###

Please comment on Professor Squarepunkt's lecture. Do you agree with him? What has he actually shown? Can you find a proof that the real numbers are uncountable in binary, or is Squarepunkt right and we will have to start all over again with the foundations. (The idea behind Squarepunkt's lecture is due to a real mathematician - Nathaniel Hellerstein.)

COMMENTARY
by LK

Professor Squarepunkt is right that there is a counterexample to the diagonal argument in binary. This does not mean that we cannot prove that the real numbers are uncountable. We can do this in various ways. Please read the discussion below.

Let the set of real numbers greater than or equal to 0 and less than 1 be denoted by **U**. Lets use the notation [a1,a2,a3,...] for the binary representation of a real number greater than or equal to 0 less than 1. Thus [1,0,1,0,1,...] = .10101...

Let ~ 0 = 1 and ~ 1 = 0. Then define

Flip[a1,a2,...] = [~ a1, ~ a2, ...].

Thus Flip[1,0,1,1,1,0,...] = [0,1,0,0,0,1,...].

Now lets think about representing numbers. We have that [1,1,1,1,...] = .111... = 1.000... and so we will not allow the sequence [1,1,1,...] since it represents 1, and we want to think about just the numbers r with r >= 0 and r <1. We have called this set **U**.

We also have that [0,1,1,1,...] = [1,0,0,0,...] since .0111... = .1000... More generally, we have
[a1,a2,...,an, 0,1,1,1,...] = [a1,a2,...,an,1,0,0,0,...].
Note that we consider infinite tails of 1's and there must be a first place where there is a 0. That is the role of the 0 in the above formula.

Two sequences of 0's and 1's represent the same real number in U if and only if they are identical term by term as sequences, or if one can be obtained from the other by eliminating an infinite tail of 1's as above. It would take us too far afield to verify this statement, but please remember that, as a real number the sequence [a1,a2,...] means the limit of the infinite sequence

$$a1/2 + a2/4 + a3/8 + \dots + an/2^n + \dots$$

where an= 0 or 1 are regarded as integers 0 or 1 and the limit is a limit of this sum of fractions.

So now lets ask: *When can Flip[a] = a as real numbers?*
We can assume that a is represented uniquely by a sequence that does not have an infinite tail of zeros.

Lemma2. *The only solution to Flip[a] = a in U is a = [1,0,0,0,...].*
Proof. Since Flip[a] differs from a in every entry as a sequence, it follows that in order for Flip[a] to equal a in U that Flip[a] must end in an infinite sequence of 1's. Thus Flip[a] must have the form
Flip[a] = [a1,a2,...,an,0,1,1,1,...] as a sequence, and so
a = [~a1,...,~an,1,0,0,0,...] as a sequence. We have that
Flip[a] = [a1,a2,...,an,0,1,1,1,...] = [a1,a2,...,an, 1,0,0,0,...].
Thus the only way that we can have Flip[a] = a is when the sequence a1,a2,...an is empty. Then we have a = [1,0,0,0,...] and
Flip[a] = [0,1,1,1,...] = [1,0,0,0,...]. **QED.**

Given any list L of real numbers from U we write them as a(1),a(2),a(3),... where each a(m) is an infinite binary string: a(n) = [a(n)1,a(n)2, a(n)3,...]. For a given list, define the diagonal D by the formula Dn = ~ a(n)n.
This defines Cantor's diagonal for any n.

From now on, assume that we are discussing lists such that every element on the list is represented by a sequence without an infinite tail of zeros. All real numbers can be represented this way by our previous discussion. We shall say that a list is *incomplete* if it does not contain every real number in U.

As we have seen, when we make the diagonal of a list L, it may happen that Flip(D) is on the list. This is Squarepunkt's example. But consider: *How can Flip(D) be on the list?* As a sequence Flip(D) differs from every sequence on the list. Thus Flip(D) on the list means that Flip(D), after elimination of a tail of ones, is on the list. (This is what happened in Squarepunkt's example.) But, in order for Flip(D) to have an infinite tail of ones, D must have an infinite tail of zeros.

Lemma3. *Let L be a list of real numbers in U such that each representative in L has no infinite tail of 1's. Let D be the diagonal of this list. If D has an infinite tail of zeros, then it is possible for Flip(D) to be on the list. In fact we have*
D= [a(1)1, a(2)2,..., a(n)n, 1, 0,0,0,0,0,0,...] for some n,
where the 1 is the first occurence in D of 1 just before the infinite tail of zeros. We then have
Flip(D) = [~a(1)1, ~a(2)2,..., ~a(n)n, 0, 1,1,1,1,1,1,...]
= [~a(1)1, ~a(2)2,..., ~a(n)n, 1,0,0,0,0,0,0,...].
One can manufacture lists where Flip(D) is on the list.
Proof. Omitted. //

Example.
a(1) = [1,0,0,0,0,0,0,...]
a(2) = [0,1,0,0,0,0,0,...]
a(3) = [0,0,1,0,0,0,0,...]
a(4) = [0,0,0,0,1,0,0,...]
a(5) = [0,0,0,0,0,1,0,...]
...
Then D=[1,1,1,0,0,0,...] and
Flip(D) = [0,0,0,1,1,1,...] = [0,0,1,0,0,0,...] = a(3).

Since we can make so many examples of infinite binary lists for which the diagonal process does not work, it makes sense to take a different approach to the problem. We already know (by a correct diagonal argument) that the set of all sequences of 0's and 1's is uncountable. U is in one-to-one correspondence with a subset of those sequences. As we have discussed above, the U-subset is the set of sequences that do not have an infinite tail of 1's. Any such sequence corresponds to a unique real number. Let S denote all possible sequences of 0's and 1's. Then we have remarked that
U = S - T where
T = all sequences with an infinite tail of 1's.

Lemma. *T is countable.*
Proof. You can list the elements of T by first listing all seqences where the tail begins immediately, then list all sequences where the tail begins at the second place, then where the tail begins at the third place and so on. Each of these is a finite list. Thus T is countable. **QED.**

Lemma. *U is uncountable.*
Proof. S is uncountable. And removing a countable set from an uncountable set leaves an uncountable set as a remainder. To see this, suppose that a set A is the disjoint union of sets B and C with both B and C countable. Then certainly A is countable. So if we know that A is uncountable and that A is the disjoint union of B and C with C countable, then it follows that B is uncountable. In this way our knowledge that S is uncountable implies that U is also uncountable.
QED.

Discussion

One can think about the Cantor diagonal argument for all the 0,1 sequences and then use the special properties of the real number sequences to single them out in one-to-one correspondence with a subset of all 0-1 sequences. We showed that the complement of this subset is countable and hence that U is uncountable. With that we see that Professor Squarepunkt has given us a very interesting class of examples, but he has not created a contradiction with Cantor's results.

On the other hand, look at Cantor's basic result:

Cantor's Theorem. For any set X, the cardinality of the power set P(X) is greater than the cardinality of X.
Proof. Let F:X ----> P(X). We prove that F is not surjective by exhibiting the set C = {x | x is in X and x is *not* in F(x)}. For if x is in F(x) then x is not in C, and if x is not in F(x) then x is in C. So no F(x) can be equal to C.
QED.

The result is so influential and the proof so short that the situation is really quite startling. Cantor's proof is the door to ever-larger sizes of infinity. Before Cantor, one would not have suspected this infinite structure to infinity. Probably many mathematicians would have assented to a notion of infinity, but not to infinitely many different sizes of infinity!

The Halting Problem

New perspectives arise as one begins to examine the form of Cantor's argument in different contexts.

For example, consider making lists of *algorithms*. You can think of an algorithm as a computer program written in some fixed language. I am concerned with algorithms A such that if you give A a natural number, then A will run for a while and eventually stop and output A(n) = 0 or 1. For different n, A(n) may have different values. The key property that we assume for A is that given an input n, A will STOP after a finite amount of time and give the value of A(n). It may take longer for some numbers n and less time for others, but it always does stop. We call A a *halting algorithm.* Can we make a list of all such algorithms?

Certainly we can make a list of all programs that look like halting algorithms. This is just a matter of writing down small programs first and larger ones later. But to check whether an algorithm can halt may be hard. You need to know how it behaves for an infinite number of inputs from N = {1,2,3,...}. So it would be good to theorize about the countability of such algorithms.

Lets suppose that we have a countable list of algorithms
A[1],A[2],A[3],...
The output of A[m] for input n will be denoted by A[m](n).

Then since the output is 0 or 1 we can use ~ 0 = 1 and ~ 1 = 0 as before. Now define a new algorithm B by the following formula
B(n) = ~ A[n](n).

We have applied the Cantor diagonal process to the list of algorithms. And the new algorithm B is certainly not on the list, since its behaviour is different from all the algorithms on the list. If the computer language for the algorithms is rich enough to express B, then we must conclude that there is no way to make a list of all the halting algorithms.

This means that in the context of a sufficiently rich language for algorithms the problem of determining whether a given program/algorithm halts is undecideable. There can be no single algorithm that takes programs as its input and decides if these programs halt. If there were such a decision procedure for halting, then we could list all the halting algorithms. Just list all grammatically correct alogorithm/programs and test each one to see if it halts with the magic halting test. Well the Cantor diagonal argument shows us that the magic test for halting does not exist. The halting problem is undecideable.

We have shown that the set of halting algorithms is uncountable since they cannot be listed. But does this mean that there are MORE of them than the natural numbers? I would say not, since after all, whatever these algorithms are they are a subcollection of all programs and the set of all programs in a given language is countable. Thus we see that the phenomenon of uncountablity can be modeled inside a countable context. This DOES mean that we should take all our talk about the SIZES of infinities with a grain of salt. The mathematical results are correct. How we think about them is open for discussion.