Another Proof

We just saw a proof that languages that are not Turing-recognizable exist based on the fact that a decider for A_{TM} cannot exist

Let's look at another proof that shows that some languages are not algorithmic.

The proof proceeds by showing that the set of all Turing machines is countably infinite whereas the set of all languages is uncountable. Therefore, there exist some languages that are not recognized by a Turing machine.

NOTE: Let $\aleph_0$ be the cardinality of the natural numbers and C the cardinality of the reals, then Cantor's continuum hypothesis states that $\aleph_0 < C$. That is, the natural numbers are *countably infinite* whereas the reals are *uncountable*.[a]

"There are fewer natural numbers than there are reals."

[a] The book has the classical proof of the uncountability of the reals based on diagonalization.

Countable Sets

Here are some simple examples of countable sets:

n	$f(n)$
1	1
2	2
$\vdots$	$\vdots$
k	k

n	$f(n)$
1	10
2	20
$\vdots$	$\vdots$
k	$k * 10$

n	$f(n)$
1	2
2	4
$\vdots$	$\vdots$
k	$k * 2$

n	$f(n)$
1	1
2	3
$\vdots$	$\vdots$
k	$k * 2 - 1$

Observation: In all cases the mapping f between n and $f(n)$ is one-to-one and onto, that is, it is bijective: Each value of n uniquely identifies a value of $f(n)$ and there is no way to construct a member of the codomain of f that does not already appear in the correspondence.

Observation: If we choose $k = \infty$ then we call the set $f(n)$ countably infinite.

The Reals are Uncountable

Proposition: The set of all reals is uncountable.

Proof: By contradiction. Assume that the set of all reals is countable. Let $f : \mathbb{N} \longrightarrow \mathbb{R}$ be a bijective mapping from the naturals to the set of all reals. Only consider the reals in the interval $[0, 1]$. Then by assumption we have,

n	$f(n)$
1	$0.3156978\ldots$
2	$0.6539879\ldots$
3	$0.1134768\ldots$
$\vdots$	$\vdots$
k	$0.2200354653\ldots5\ldots$
$\vdots$	$\vdots$

That is, since f is a bijective mapping we have a list of all possible real values. We now show that we can construct a real value that is not included in the list above. We construct this value by taking the ith digit after the decimal point for each real value identified by i appearing in the correspondence and adding one to it (modulo 10). We use these newly generated digits to construct a new value in $[0, 1]$: $0.464\ldots6\ldots$

By construction this value differs from any real value appearing in the codomain of the mapping by at least one digit. This means that there is at least one value in the codomain of f which is not in the image of f. This is a contradiction, since f was assumed to be a bijective mapping. Therefore, our assumption that the set of all reals is countable must be wrong. □

Diagonalization

The general proof technique is as follows:

1. Assume that you have a correspondence $f : \mathbb{N} \rightarrow \mathcal{S}$, where $\mathcal{S}$ is the structure you want to investigate.
2. Construct a grid with the rows containing elements of your correspondence.
3. Now construct a new item of your structure as an element of $\mathcal{S}$ such that it will differ from all other elements in the correspondence on the major diagonal.
4. This is a contradiction since you have constructed an element in $\mathcal{S}$ not listed in the correspondence, therefore, f is not a correspondence and the structure $\mathcal{S}$ is not countable.

Turing Machines are Countably Infinite

Theorem: The set of all Turing machines is countably infinite.

Proof: Let $\Sigma_{0,1} = \{0, 1\}$ be the alphabet over the symbols 0 and 1, observe that the set of all strings over this alphabet, say $\Sigma^*_{0,1}$, is countably infinite by the fact that we can interpret each string in this set as the binary encoding of a natural number. Now, let $\langle M \rangle_{0,1}$ be the binary encoding of some Turing machine M. It is clear that such an encoding exists, since any other encoding $\langle M \rangle_{\Sigma'}$ can be transformed into the encoding $\langle M \rangle_{0,1}$ by representing each symbol in Σ' as a unique string in $\Sigma_{0,1}$.[a] Now, let $\langle M \rangle^*_{0,1}$ be the set of all encoded, valid Turing machine descriptions. Observe that $\langle M \rangle^*_{0,1} \subseteq \Sigma^*_{0,1}$. This implies that the set of all Turing machines is countable infinite.□

[a] We do this everyday on our digital computers.

Infinite Binary Sequences are Uncountable

Theorem: The set of all infinite binary sequences is uncountable.

Proof: We prove this by contradiction using the diagonalization method. Assume that we can construct a bijective mapping $f : \mathbb{N} \rightarrow \mathcal{B}$, where $\mathbb{N}$ are the natural numbers and $\mathcal{B}$ is the set of all infinite binary sequences. Then,

n	$f(n)$
1	$0100111\ldots$
2	$11111000\ldots$
3	$1011001\ldots$
$\vdots$	$\vdots$
k	$0010\ldots 0_k \ldots$
$\vdots$	$\vdots$

Observe that we can always construct another binary sequence which will differ from all the enumerated sequences by at least one bit,

$$100\ldots 1_k \ldots$$

That is, for any value i we have constructed a sequence which will differ in value from $f(i)$ in the i^{th} bit. Therefore, there exist elements in $\mathcal{B}$ that are not in the image of f. That means our assumption that f is bijective is incorrect. □

Languages are Uncountable

Theorem: The set of all languages L over alphabet $\Sigma_{0,1}$ is uncountable.

Proof: We show this by constructing a bijective mapping $f : L \rightarrow \mathcal{B}$. For each language $A \in L$ we can construct a unique element in $\mathcal{B}$ called the *characteristic sequence*. Let $\Sigma_{0,1}^* = \{s_1, s_2, s_3, \ldots\}$, then the ith bit of the characteristic sequence of A is 1 if $s_i \in A$ and 0 if $s_i \notin A$. Note,

- The empty language has the characteristic sequence $000000\ldots$
- The language $\Sigma_{0,1}^*$ has the characteristic sequence $1111\ldots$

The mapping f is bijective in that any possible language in L has a unique sequence in $\mathcal{B}$ and any sequence in $\mathcal{B}$ uniquely defines a language in L. □

Not Turing-recognizable Languages

Theorem: Some languages are not Turing-recognizable.

Proof: Observe that $\#\langle M \rangle^*_{0,1} \leq \aleph_0$ and $\aleph_0 < \#L$. It follows from previous proofs that there are some languages that are not recognized by a Turing machine. $\square$

"There are more languages than there are Turing Machines."