Lecture #13: First Hard and Undecidable Languages
Lecture Presentation

Preliminaries: Listing Various Kinds of Infinite Sets

Countable Sets

Let $\mathbb{N} = \{0, 1, 2, 3, \dots\}$ be the set of non-negative integers.

A set S is ***countable*** is there is a total function $f : \mathbb{N} \to S$ that is ***surjective***, that is "onto": For every element x of S there exists a non-negative integer n such that $f(n) = x$.

- Any non-empty ***finite*** set
$$S = \{x_1, x_2, \dots, x_k\}$$
is countable: Let $f : \mathbb{N} \to \mathbb{N}$ such that, for every non-negative integer n,
$$f(n) = \begin{cases} x_{n+1} & \text{if } 0 \leq n \leq k-1 \\ x_k & \text{if } n \geq k. \end{cases}$$
This is a well-defined total function from $\mathbb{N}$ to S. To see that it is surjective, let $x \in S$. Then $x = x_i$ for some integer i such that $1 \leq i \leq k$, and $f(i-1) = x_i = x$. Since x was arbitrarily chosen from S it follows that f is surjective (and S is countable.

As the examples to follow show, some (but not all) infinite sets are countable, as well.

Countability of the Set of Strings over an Alphabet

Consider an alphabet

$$\Sigma = \{\sigma_1, \sigma_2, \dots, \sigma_k\}$$

- For every non-negative integer n, the number of strings in $\Sigma^\star$, with length n, is k^n.
- For every non-negative integer n, the number of strings in $\Sigma^\star$, with length *at most* n is

$$\mu(n) = \sum_{i=0}^{n} k^i = \frac{k^{n+1} - 1}{k - 1} \tag{1}$$

  — using a formula for the closed form of a ***geometric series*** that you have, ideally, seen before.

- Consider a map $\rho : \Sigma \to \mathbb{N}$ such that $\rho(\sigma_i) = i - 1$ for every integer i such that $1 \leq i \leq k$. Then

$$\{j \in \mathbb{N} \mid j = \rho(\alpha) \text{ for a symbol } \alpha \in \Sigma\} \\ = \{j \in \mathbb{N} \mid 0 \leq j \leq k - 1\} = \{0, 1, 2, \dots, k - 1\}.$$

- This can be extended to obtain a mapping ρ_n from the set of strings in $\Sigma^\star$ with length n, to $\mathbb{N}$, by setting

$$\begin{aligned}\rho_n(\alpha_1 \alpha_2 \dots \alpha_n) &= \sum_{i=1}^{n} \rho(\alpha_i) \cdot k^{n-i} \\ &= \rho(\alpha_1) \cdot k^{n-1} + \rho(\alpha_2) \cdot k^{n-2} + \cdots + \rho(\alpha_{n-1}) \cdot k + \rho(\alpha_n).\end{aligned}$$

  Suppose, for example, that $\Sigma = \{0, 1\} = \{\sigma_1, \sigma_2\}$ (where $\sigma_1 = 0$ and $\sigma_2 = 1$) — so that $\rho(0) = \rho(\sigma_1) = 0$ and $\rho(1) = \rho(\sigma_2) = 1$. If $n = 3$ then this defines a mapping $\rho_3)$ such that $\rho_3(000) = 0$, $\rho_3(001) = 1$, $\rho_3(010) = 2$, $\rho_3(011) = 3$, $\rho_3(100) = 4$, $\rho_3(101) = 5$, $\rho_3(110) = 6$, and $\rho_3(111) = 7$.

  A Useful Property: In general, if $|\Sigma| = k$ as above, and $n \in \mathbb{N}$ then, for every integer i such that $0 \leq i \leq k^n - 1$, there is ***exactly*** one string $\omega \in \Sigma^\star$ such that $|\omega| = k$ and $\rho_k(\omega) = i$.

- Consider a mapping $\widehat{\rho} : \Sigma^\star \to \mathbb{N}$ such the following properties are satisfied:
  - (i) $\widehat{\rho}(\lambda) = 0$.
  - (ii) For every *positive* integer n, and for every string $\omega \in \Sigma^\star$ such that $|\omega| = n$,

$$\widehat{\rho}(\omega) = \mu(n-1) + \rho_n(\omega). \tag{2}$$

Once again, consider the alphabet $\Sigma = \{0, 1\}$ (where $\sigma_1 = 0$ and $\sigma_2 = 1$) as above. The values $\widehat{\omega}(\omega)$, for every string $\omega \in \Sigma^\star$ such that $|\omega| \leq 3$, is as shown in the following table.

ω	$n = \lvert\omega\rvert$	$\mu(n-1)$	$\rho_n(\omega)$	$\widehat{\rho}(\omega)$
λ				0
0	1	1	0	1
1	1	1	1	2
00	2	3	0	3
01	2	3	1	4
10	2	3	2	5
11	2	3	3	6
000	3	7	0	7
001	3	7	1	8
010	3	7	2	9
011	3	7	3	10
100	3	7	4	11
101	3	7	5	12
110	3	7	6	13
111	3	7	7	14

Now, since $\mu(3) = 15$ one can also see that $\widehat{\rho}(0000) = 15 = \widehat{\rho}(111) + 1$.

It is possible to prove — for *every* alphabet Σ — that the function $\widehat{\rho} : \Sigma^\star \to \mathbb{N}$ is an ***bijective*** function: For every non-negative integer ℓ, there is ***exactly one*** string $\omega_\ell \in \Sigma^\star$ such that $\widehat{\rho}(\omega_\ell) = \ell$.

Continuing this example, one sees that that, for $\Sigma = \{0, 1\}$, $\omega_0 = \lambda$, $\omega_1 = 0$, $\omega_2 = 1$, $\omega_3 = 00$ — and the strings ω_ℓ for listed, for increasing ℓ, by continuing down the rows of the table.

Since the function $\widehat{\rho}$ is injective, it has a well-defined ***inverse function***, namely, a function $f : \mathbb{N} \to \Sigma^\star$ such that $f(\widehat{\rho}(\omega) = \omega$ for every string $\omega \in \Sigma^\star$ and $\widehat{\rho}(f(\ell)) = \ell$ for every non-negative integer ℓ. The function f is certainly surjective (since it is also "injective") — is needed to establish that — for every alphabet Σ — the set $\Sigma^\star$, of all strings over Σ, is a ***countable*** set.

What Does This “Listing” of Strings in $\Sigma^\star$ Formalize?

Application for Turing Machines

Consider the set of Turing machines — as given by strings in the language TM $\subseteq \Sigma^{\star}_{\mathsf{TM}}$.

One can show that the set of Turing machines is a countable set — and describe a way to *list* all Turing machines in a sequence

$$M_0, M_1, M_2, M_3, \ldots$$

(where each Turing machine could be listed more than once, but is always listed *at least* once), as follows:

One can also show that the set of Turing machines with the form

$$M = (Q, \Sigma, \Gamma, \delta, q_0, q_{\text{accept}}, q_{\text{reject}})$$

such that $\Sigma = \{0, 1\}$ (that is, $|\Sigma| = 2$) is a countable set — and describe a way to *list* all such Turing machines

$$\widehat{M_0}, \widehat{M_1}, \widehat{M_2}, \widehat{M_3}, \ldots$$

(where every such Turing machine could be listed more than once, but is always listed *at least* once), as follows:

What This Gives Us

Claim. *There exists a language $L \subseteq \Sigma^\star$, where $\Sigma = \{0, 1\}$, such that L is unrecognizable.*

Proof: By contradiction. Let us ***assume*** that every language $L \subseteq \Sigma^\star$, where $\Sigma = \{0, 1\}$, is recognizable. Then...

What Else Can We Establish Using This Idea?

Why is This Not Sufficient — Why Do We Need the Result in the Notes, Too?