Lecture #15: More about Reductions and Reducibilities
Lecture Presentation

One-One Reducibility

Definition for Computations with $\mathcal{S}$-Programs

Let $A, B \subseteq \mathbb{N}$. Then A is ***one-one reducible*** to B, "$A \preceq_1 B$", if there exists a one-one computable total function $f : \mathbb{N} \rightarrow \mathbb{N}$ such that

$$A = \{x \in N \mid f(x) \in B\}.$$

Moving to Other Models — When Defining a Relation Between Subsets of $\mathbb{N}$

Moving to Other Models — When Defining a Relation Between Languages

Many-One Reducibility

Definition for Computations with $\mathcal{S}$-Programs

Let $A, B \subseteq \mathbb{N}$. Then A is ***many-one reducible*** to B, "$A \preceq_m B$", if there exists a computable total function $f : \mathbb{N} \to \mathbb{N}$ such that

$$A = \{x \in N \mid f(x) \in B\}.$$

Moving to Other Models

Turing Reducibility

Definition for Computations with $\mathcal{S}$-Programs

Let $G : \mathbb{N} \to \mathbb{N}$ be a total function. A partial or total function $f : \mathbb{N} \to \mathbb{N}$ is ***G-computable*** if there is an $\mathcal{S}$-program $\mathcal{P}$ (with an oracle for G) that G-computes f.

Let $A, B \subseteq \mathbb{N}$. Then A is ***Turing-reducible*** to B, "$A \preceq_T B$", if the characteristic function p_A, of A, is p_B-computable, where p_B is the characteristic function of B.

Moving to Computations with $\mathcal{S}_n$-Programs

Moving to Computations with Post-Turing Programs

Moving to Computations with Turing Machines

Conclusions