Lecture #11: Reducibilities — and Many-One Reductions

Lecture Presentation

Getting Started

Consider a function $p : \mathbb{N} \to \mathbb{N}$ such that, for every natural number n, $p(n)$ is the index of the $\mathcal{S}$-program obtained from the following one using macro expansion[1].

$$
\begin{array}{rl}
 & Z_1 \leftarrow \Phi^{(1)}(0, n) \\
 & Z_2 \leftarrow \Phi^{(1)}(X_1, n) \\
 & Z_3 \leftarrow Z_1 - Z_2 \\
[A_1] & \text{IF } Z_3 \neq 0 \text{ GOTO } A_1 \\
 & Z_4 \leftarrow Z_2 - Z_1 \\
[A_2] & \text{IF } Z_4 \neq 0 \text{ GOTO } A_2
\end{array}
$$

(a) Describe the partial or total function that is, the partial or total function computed by the $\mathcal{S}$-program whose encoding is $p(n)$): What is the value of this function at an integer i?

[1] That is, $p(n)$ should be the index of a function that computes the same partial function as the program shown here.

(b) Use a result, established in recent lectures, that can be used to give a reasonably short and simple function that the function $p : \mathbb{N} \to \mathbb{N}$ is ***primitive recursive***.

A Many-One Reducibility

Recall, from the lecture notes, that

$$\text{Total} = \{n \in \mathbb{N} \mid \text{the } \mathcal{S}\text{-program } \mathcal{P}\text{, such that } \#(\mathcal{P}) = n\text{, computes a } \textbf{\textit{total}} \text{ function}\}.$$

Let

$$\text{Constant} = \{n \in \mathbb{N} \mid \text{the } \mathcal{S}\text{-program } \mathcal{P}\text{, such that } \#(\mathcal{P}) = n\text{, computes a } \textbf{\textit{constant}} \text{ function}\}.$$

(a) Note that Constant $\subseteq$ Total, since every constant function is a total function. Does this imply that Constant $\preceq_m$ Total? Explain ***briefly*** why your answer is correct.

(b) Let $n \in \mathbb{N}$ such that $n \in$ Constant. Prove that $p(n) \in$ Total, where $p : \mathbb{N} \to \mathbb{N}$ is the primitive recursive function considered in the previous question.

(c) Let $n \in \mathbb{N}$ such that $n \notin$ Constant. Prove that $p(n) \notin$ Total.

(d) Using the above as needed, prove that Constant $\preceq_m$ Total.

Another Many-One Reduction

Now prove that Total $\preceq_m$ Constant, as well.

Try to use the same strategy as the one used to prove that Constant $\preceq_m$ Total. That is, identity a function $q : \mathbb{N} \to \mathbb{N}$ which can be thought of as mapping encodings of $\mathcal{S}$-programs to encodings of related $\mathcal{S}$-programs. Use a result from lectures to prove that q is a primitive recursive function. Then use q to prove that Constant $\preceq_m$ Total, as claimed.

Conclusions

What can now be concluded about the relationship between the sets Total and Constant?