Sections 5.1–5.3: Reducibility


**Definitions**

Define each of the following concepts:

(a) Decidable language

(b) $A_{TM}, E_{TM}, EQ_{TM}, HALT(onThisInput)_{TM}, HALT(onAllInputs)_{TM}$

(c) Computation history

(d) Linear bounded automaton, $LBA$

(e) Post Correspondence Problem

(f) Mapping reducibility

(g) A reduction of $A$ to $B$

(h) Computable function

**Results**

The number of distinct configurations of a linear bounded automaton with $q$ states, tape length $n$, and $g$ symbols in the tape alphabet is $qng^n$.

If $A \leq_m B$ and $B$ is decidable, then $A$ is decidable.

If $A \leq_m B$ and $A$ is undecidable, then $B$ is undecidable.

If $A \leq_m B$ and $B$ is Turing-recognizable, then $A$ is Turing-recognizable.

If $A \leq_m B$ and $A$ is not Turing-recognizable, then $B$ is not Turing-recognizable.

**The Frontier of Computability**

Create a matrix displaying a two-way classification of a collection of languages, characterizing them by machine type along one dimension

$TM, LBA, PDA, FSA,$

and by problem type along the other dimension

$A, E, EQ, INF, ALL, HALT(onThisInput), HALT(onAllInputs).$

Thus, the language $A_{TM}$ would appear in the row corresponding to Turing Machines, $TM$, and in the column corresponding to problem type $A$. Classify each of the languages in your display as decidable or
undecidable, and cite a theorem, corollary, exercise, or a proof of your own devising as the relevant authority. Draw a contour, or contours, through your matrix indicating the frontiers of decidability. Add additional languages which don’t quite fit into the above scheme into an area off to the side, and and use that area to illustrate interesting patterns such as pairs of languages and their complements having different classifications, languages which are recognizable but not decidable, languages which are not even recognizable, and the like.

### Computable Functions

(3 Points) Define the characteristic function, $\chi_L$, of a language, $L$, by

$$\chi_L(w) = \begin{cases} 
1 & \text{if } w \in L, \\
0 & \text{otherwise.}
\end{cases}$$

(a) Define the notion of a computable function.

(b) Prove or disprove:

$L$ is a decidable language $\iff$ The characteristic function, $\chi_L$, of $L$ is a computable function.

### Exercises

We will attempt to solve each of the following exercises as a community project in class today. Finish these solutions as homework exercises, write them up carefully and clearly, and hand them in at the beginning of the next class.

*Exercises for Chapter 5, page 195*: 1, 2, 3, 4, 5, 6, 7, 8

*Problems for Chapter 5, pages 169–170*: 9, 11, 12, 22, 23