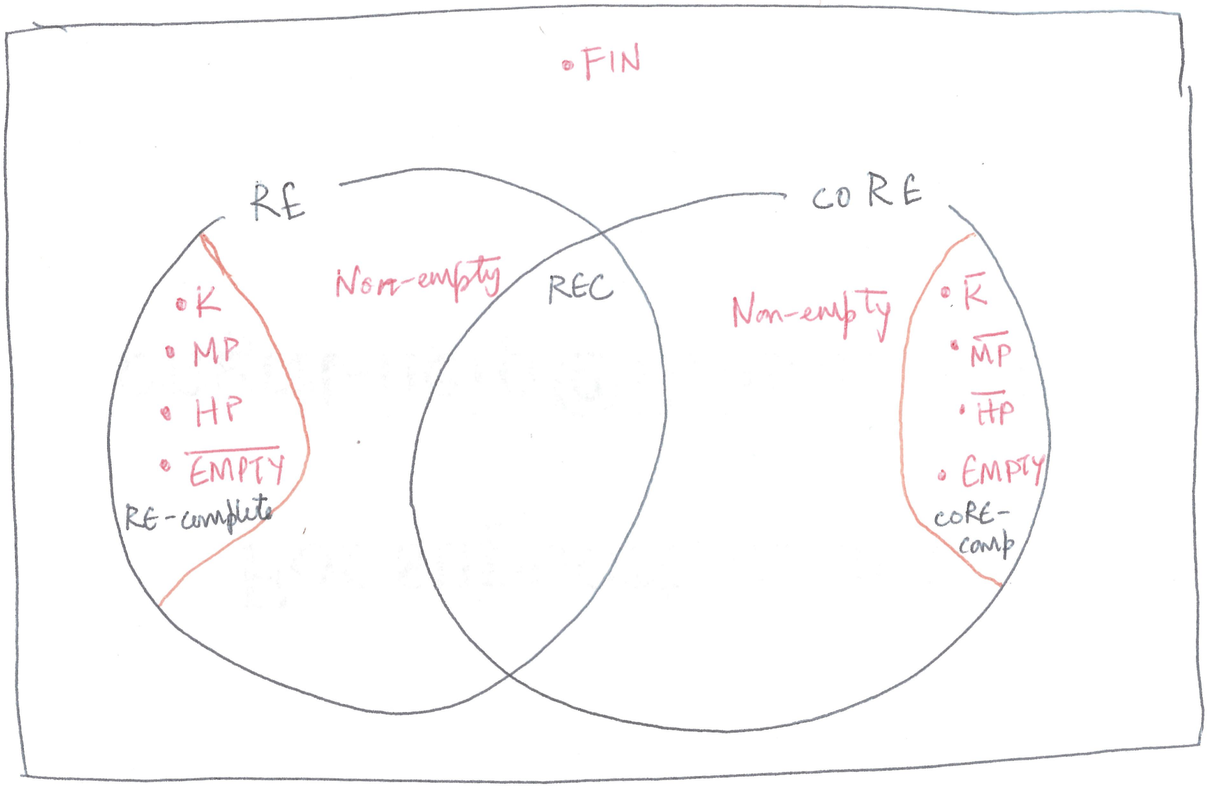

$K \leq_m MP$, $MP \leq_m HP$, $\overline{K} \leq EMPTY$, $\overline{K} \leq_m FIN$, $\overline{K} \leq_m \overline{FIN}$

Lecture 12: Oracle Turing Machines and Arithmetic Hierarchy

Date: October 3, 2023.

Oracle Turing Machine is a 2-tape Turing machine $M = (Q, \Sigma, \Gamma, \vdash, \sqcup, \delta, s, t, r)$. The first tape is the regular tape of the Turing machine which initially holds the input, and the machine can read/write from. The second tape is called the **query tape**. The Oracle Turing machine also has 3 special states — the query state $q_?$, and the answer states q_{yes} and q_{no}.

An oracle Turing machine is executed with a language B; B is said to be the **oracle**. The machine M executed with oracle B is denoted as M^B. Such a machine proceeds like a normal (2-tape) Turing machine, until it reaches the query state $q_?$. Let x be the string written on its query tape at this point. In the next step, M^B moves to state q_{yes} if $x \in B$, and to state q_{no} if $x \notin B$. The computation of M^B then proceeds as normal, until the next point in time when it reaches the query state $q_?$. This process keeps repeating until the machine halts in t or r.

Definition 1. Given a language B, [A is r.e (recursive) iff A is r.e (rec) in $\emptyset$

- A is **recursively enumerable in** B if there is an oracle Turing machine M such that $A = \mathbf{L}(M^B)$.
- A is **recursive in** B if there is a **total** oracle Turing machine M such that $A = \mathbf{L}(M^B)$. When A is recursive in B, we also say that A **Turing reduces** to B, and is denoted as $A \leq_T B$.

Proposition 1. *If $A \leq_T B$ and $B \leq_T C$ then $A \leq_T C$.*

$\exists$ total TM M_1 s.t $A = L(M_1^B)$. $\exists$ total TM M_2 s.t $B = L(M_2^C)$

Goal: Design total TM M s.t $A = L(M^C)$.

$\rightarrow$ M: Run M_1 and whenever M_1 asks a query, it will run M_2 to answer that query.

Proposition 2. *If $A \leq_T B$ then $\overline{A} \leq_T B$.* –[If A is recursive then $\bar{A}$ is recursive

$\exists$ total TM M s.t $A = L(M^B)$

To solve $\bar{A}$: Run M with oracle B and flip answer.

Proposition 3. *$A \leq_T B$ iff A and $\overline{A}$ are both recursively enumerable in B.* –[A is recursive iff A and $\bar{A}$ are r.e.

($\Rightarrow$) $\exists$ total TM M s.t $A = L(M^B)$. $\Rightarrow$ A is r.e in B.

$\exists$ total TM $\overline{M}$ s.t $\bar{A} = L(\overline{M}^B)$ $\Rightarrow$ $\bar{A}$ is r.e in B

Due prop. 2.

($\Leftarrow$) Let M_1 and M_2 be TM s.t $A = L(M_1^B)$ and $\bar{A} = L(M_2^B)$

New Algo: Run M_1 & M_2 "in parallel".

Arithmetic Hierarchy is a hierarchy of classes inductively defined as follows.

$$\begin{array}{lcl} \Sigma^0_1 & = & \text{RE} \\ \Delta^0_1 & = & \text{REC} \\ \Sigma^0_{n+1} & = & \{\mathbf{L}(M^B) \mid B \in \Sigma^0_n\} \\ \Delta^0_{n+1} & = & \{\mathbf{L}(M^B) \mid B \in \Sigma^0_n,\ M^B \text{ is total}\} = \{A \mid \exists B \in \Sigma^0_n,\ A \leq_T B\} \\ \Pi^0_n & = & \{A \mid \overline{A} \in \Sigma^0_n\} \end{array}$$

$\Pi^0_1 = \text{coRE}$.

Observe that $\Pi^0_1 = $ co-RE.

[REC = RE $\cap$ coRE
$\Delta^0_1 = \Sigma^0_1 \cap \Pi^0_1$.

Proposition 4. *Prove that for all $n \geq 1$, $\Delta^0_n = \Sigma^0_n \cap \Pi^0_n$.*

$\Delta^0_n \subseteq \Sigma^0_n \cap \Pi^0_n$: $A \in \Delta^0_n$. $\exists B \in \Sigma^0_{n-1}$, $A \leq_T B$.

A is r.e in B and $\bar{A}$ is r.e in B $\Rightarrow A \in \Sigma^0_n, \bar{A} \in \Sigma^0_n$.
$\Rightarrow A \in \Sigma^0_n, A \in \Pi^0_n$.

$\Sigma^0_n \cap \Pi^0_n \subseteq \Delta^0_n$: $A \in \Sigma^0_n \cap \Pi^0_n$. $\exists$ TM M_1, M_2, and $B_1, B_2 \in \Sigma^0_{n-1}$. s.t.
$A = L(M_1^{B_1})$ and $\bar{A} = L(M_2^{B_2})$ [$B_1 \cap B_2$.
$B = \{0x \mid x \in B_1\} \cup \{1x \mid x \in B_2\}$ ($\in \Sigma^0_{n-1}$ to be proved) $B_1 \times B_2$
"Run M_1 & M_2 in parallel w.r.t B".

Proposition 5. *Prove that $A \in \text{RE}$ iff there is a recursive relation R such that $A = \{x \mid \exists y.\ R(x,y)\}$.*

($\Rightarrow$) $A \in RE$. $\exists$ ~~T~~M M s.t $A = L(M)$.
$R_M = \{(x,t) \mid M$ accepts x within t steps$\}$.
R_M recursive : "Run M on x for t steps."
$A = \{x \mid x$ is accepts by $M\} = \{x \mid \exists t.\ \underbrace{x \text{ is accepted by } M \text{ in } t \text{ steps}}_{R(x,t)}\}$

($\Leftarrow$) R is recursive i.e $\exists$ Total TM M s.t $R = L(M)$
$A = \{x \mid \exists y\ R(x,y)\}$
Algo for A : Input x.
for every string y.
Run M on (x,y)

Theorem 6. *1. A set $A \in \Sigma^0_n$ iff there is a recursive relation R such that*

$$A = \{x \mid \exists y_1 \forall y_2 \exists y_3 \cdots Q y_n R(x, y_1, y_2, \ldots y_n)\}$$

where $Q = \exists$ if n is odd, and $Q = \forall$ if n is even.

2. A set $A \in \Pi^0_n$ iff there is a recursive relation R such that

$$A = \{x \mid \forall y_1 \exists y_2 \forall y_3 \cdots Q y_n R(x, y_1, y_2, \ldots y_n)\}$$

where $Q = \forall$ if n is odd, and $Q = \exists$ if n is even.

Problem 1. Prove that EMPTY $= \{\langle M\rangle \mid \mathbf{L}(M) = \emptyset\} \in \Pi_1^0$.

Need a Recursive relation R s.t EMPTY $= \{x \mid \forall y\, R(x, y)\}$

$R = \{(x, y, t) \mid M_x$ does not accept y within t steps$\}$

$x \in$ EMPTY iff $\forall y \forall t\; R(x,y,t)$

$\forall \langle y,t\rangle\; R(x,y,t)$

Problem 2. Prove that TOTAL $= \{\langle M\rangle \mid M$ is total$\} \in \Pi_2^0$.

Need a recursive relation R s.t. TOTAL $= \{x \mid \forall y \exists z\; R(x,y,z)\}$

Take $R = \{(x,y,z) \mid M_x$ halts in z steps on input $y\}$.

TOTAL $= \{x \mid \forall y \exists z\; R(x,y,z)\}$.

Problem 3. Prove that FIN $= \{\langle M\rangle \mid \mathbf{L}(M)$ is finite$\} \in \Sigma_2^0$.

Problem 4. Prove that COF $= \{\langle M\rangle \mid \mathbf{L}(M)$ is cofinite$\} \in \Sigma_3^0$.

Theorem 7. *The arithmetic hierarchy is strict. That is, for every n, (a) Σ_n^0 and Π_n^0 are incomparable with respect to set inclusion, (b) $\Sigma_n^0 \cup \Pi_n^0 \subsetneq \Delta_{n+1}^0$.*

Definition 2. A language A is $\mathcal{C}$-hard, for a class $\mathcal{C}$ of languages, with respect to $\leq_m$ if for every $B \in \mathcal{C}$, $B \leq_m A$. And A is $\mathcal{C}$-complete if $A \in \mathcal{C}$ and A is $\mathcal{C}$-hard.

Proposition 8. *1.* EMPTY *is* Π^0_1*-complete.*

2. TOTAL *is* Π^0_2*-complete.*

3. FIN *is* Σ^0_2*-complete.*

4. COF *is* Σ^0_3*-complete.*