Week 8

HW3 Due Today!

HW4 (Last one!) OUT

Week 7 Summary (2 weeks ago!)

1. We saw $D = \{x \mid \{x\}_1(x) \text{ does not accept}\}$ is not r.e. by diagonalization

2. Using reductions we proved K, Halt are not recursive

Using Reductions to show other (more natural) languages / functions are not computable / recursive / r.e.

High Level:

① Say we know L_1 not recursive
To show L_2 not recursive, design a TM M_1 always halts & $\mathcal{L}(M_1) = L_1$, assuming a TM M_2 that always halts & $\mathcal{L}(M_2) = L_2$

② Suppose L_1 not r.e.
To show L_2 not r.e., construct M_1 st $\mathcal{L}(M_1) = L_1$ assuming a TM M_2 st $\mathcal{L}(M_2) = L_2$

The Halting Problem is Not Recursive

$$K \stackrel{d}{=} \{ x \mid TM\ \{x\} \text{ halts on input } x \}$$

$$HALT \stackrel{d}{=} \{ \langle x, y \rangle \mid TM\ \{x\} \text{ halts on input } y \}$$

Theorem. HALT, K are both r.e.,

Neither are recursive

The Halting Problem is Not Recursive

$$K \stackrel{d}{=} \{ x \mid TM\ \{x\} \text{ halts on input } x \}$$

Theorem K is not recursive

If K recursive then D also recursive

Theorem Halt Not recursive

If Halt recursive then K recursive

Tips

(1.) Try obvious algorithms to see if you think language is recursive, re, or neither

(2.) To show L not r.e., sometimes it helps to work with $\bar{L}$
(ie. if $\bar{L}$ r.e., + $\bar{L}$ not recursive then L not r.e.)

(3) get reduction in correct direction.
Many times constructed TM M_1 will ignore its own input

$L = \{ x \mid \{x\} \text{ accepts at least one input} \}$

$L = \{ x \mid \{x\} \text{ accepts at least one input} \}$

- L is r.e. (Dovetailing)
- L is Not recursive

$L_1 = K = \{ y \mid \{y\}(y) \text{ halts} \}$

Assume $L_2 = L$ is recursive & Let M_2 be TM $\mathcal{L}(M_2) = L$
and M_2 always halts

M_1 on input y:

Construct encoding z of TM $\{z\}$ where
$\{z\}$ on input x: Ignores x & runs $\{y\}$ on y
and accepts x if $\{y\}(y)$ halts

Run M_2 on z and accept y iff $M_2(z)$ accepts

<u>Claim</u> $\mathcal{L}(M_1) = K$ and M_1 always halts

$y \in K \Rightarrow \{y\}(y)$ halts $\Rightarrow \{z\}$ accepts all inputs $\Rightarrow M_2(z) = 1 \Rightarrow M_1(y) = 1$

$y \notin K \Rightarrow \{y\}(y)$ doesn't halt $\Rightarrow \{z\}$ accepts No input $\Rightarrow M_2(z) \neq 1 \Rightarrow M_1(y) \neq 1$

Completeness

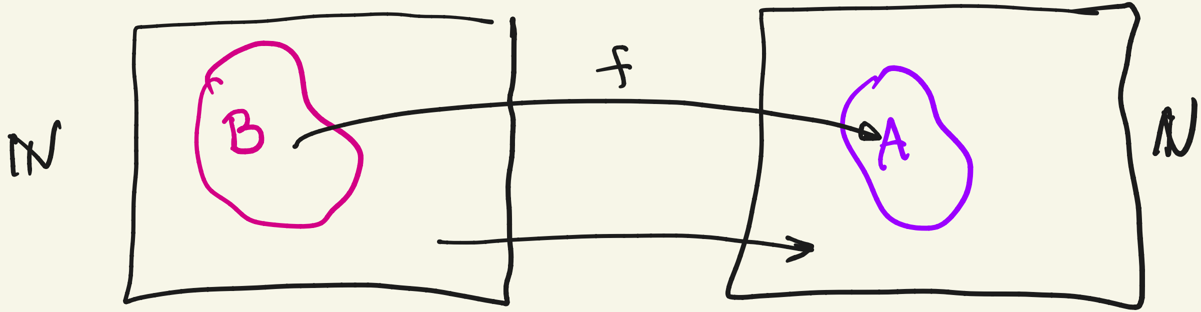
A set $A \subseteq \mathbb{N}$ is r.e.-complete if

(1) A is r.e.

(2) $\forall B \subseteq \mathbb{N}$, if B is r.e. then $B \leq_m A$

B reduces to A

so if A is recursive then B recursive



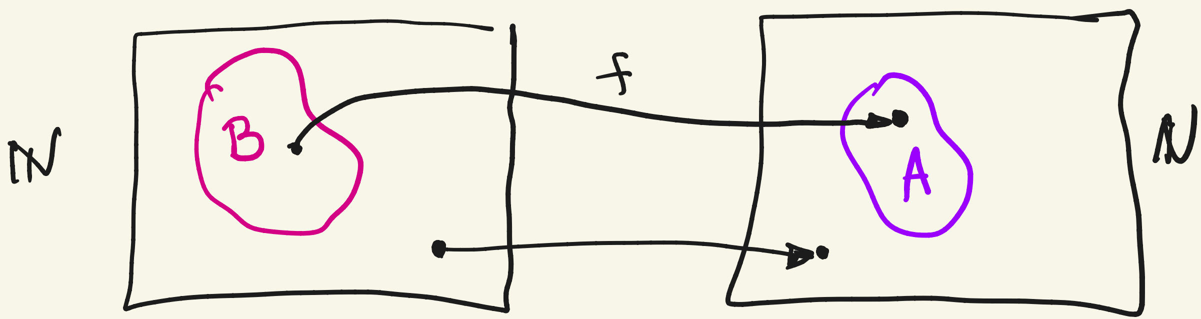
Completeness

A set $A \subseteq \mathbb{N}$ is r.e.-complete if

(1) A is r.e.

(2) $\forall B \subseteq \mathbb{N}$, if B is r.e. then $B \leq_m A$

$\exists$ computable function $f : \mathbb{N} \Rightarrow \mathbb{N}$ such that
$\forall x \quad f(x) \in A \Leftrightarrow x \in B$



Hilbert's 10^{th} Problem (1900)

A diophantine equation is of the form $p(\vec{x}) = 0$ where p is a polynomial over variables $x_1, \dots, x_n$ with integer coefficients

Ex $3x_1^5 x_2^3 + (x_1+1)^8 - x_7^{10} = 0$

$\mathcal{L}_{DIOPH} = \{ \langle p \rangle \mid p \text{ has a solution over } \mathbb{N} \}$

Theorem

$\mathcal{L}_{DIOPH}$ is r.e.-complete

An Equivalent Characterization of RE Sets

Let $f : \mathbb{N} \to \mathbb{N}$

Then $R_f \subseteq \mathbb{N} \times \mathbb{N}$

is the set of all pairs (x, y) such that $f(x) = y$

* **Theorem** f computable if and only if R_f is r.e.

An Equivalent Characterization of RE Sets

Let $f : \mathbb{N} \to \mathbb{N}$

Then $R_f \subseteq \mathbb{N} \times \mathbb{N}$

is the set of all pairs (x, y) such that $f(x) = y$

* **Theorem** f computable if and only if R_f is r.e.

Proof $\Rightarrow$: Suppose f computable.

TM for R_f on input (x,y):

Run TM computing f on x.

If it halts and outputs y then accept (x,y)

Otherwise reject (x,y)

An Equivalent Characterization of RE Sets

Let $f : \mathbb{N} \to \mathbb{N}$

Then $R_f \subseteq \mathbb{N} \times \mathbb{N}$

is the set of all pairs (x, y) such that $f(x) = y$

* **Theorem** f computable if and only if R_f is r.e.

Proof $\Leftarrow$: Let R_f be r.e. with TM M

On x: Enumerate all $\mathbb{N}$: $y_1, y_2, \ldots$

For $i = 1, 2, \ldots$

For all $j \leq i$: simulate M on (x, y_j) for i steps

If simulation accepts (x, y_j),

halt & output y_j

A second Characterization of RE Sets

***Theorem** A relation $A \subseteq \mathbb{N}^k$ is r.e.
if and only if there is a recursive relation
$R \subseteq \mathbb{N}^{k+1}$ such that

$$\vec{x} \in A \iff \exists y\, R(\vec{x}, y) \quad \forall \vec{x} \in \mathbb{N}^n$$

Note We defined A to be r.e. iff there is a TM M
such that $\forall \vec{x} \in \mathbb{N}^n$ $\left(M(\langle x \rangle) \text{ accepts} \iff \vec{x} \in A \right)$

A Second Characterization of RE Sets

*** Theorem** A relation $A \subseteq \mathbb{N}^k$ is r.e.
if and only if there is a recursive relation
$R \subseteq \mathbb{N}^{k+1}$ such that

$$\vec{x} \in A \iff \exists y\, R(\vec{x}, y) \quad \forall \vec{x} \in \mathbb{N}^n$$

Proof sketch

$\Rightarrow$: Let A be r.e., $\mathcal{L}(M) = A$

$R(\vec{x}, y)$: view y as encoding of an $m \times m$ tableaux for some $m \in \mathbb{N}$

$(\vec{x}, y) \in R \iff M(\vec{x})$ halts in m steps and accepts and y is the $m \times m$ tableaux of $M(\vec{x})$

A second Characterization of RE Sets

***Theorem** A relation $A \subseteq \mathbb{N}^k$ is r.e. if and only if there is a recursive relation $R \subseteq \mathbb{N}^{k+1}$ such that

$$\vec{x} \in A \iff \exists y\, R(\vec{x}, y) \quad \forall \vec{x} \in \mathbb{N}^n$$

Proof sketch

$\Leftarrow$ Let $R \subseteq \mathbb{N}^{k+1}$ be recursive relation such that $\vec{x} \in A \iff \exists y\, R(\vec{x}, y)$, & Let $\mathcal{L}(M) = R$

on input $\vec{x}$:
  For $i = 1, 2, \ldots$
    For $j = 1$ to i
      Run M on $(\vec{x}, y_j)$
      halt & accept if $M(\vec{x}, y_j)$ accepts

Review of Definitions

$\mathcal{L}_A = \{0, s, +, \cdot\, ; =\}$ Language of arithmetic

$\Phi_0 =$ all $\mathcal{L}_A$-sentences

$TA = \{A \in \Phi_0 \mid \mathbb{N} \models A\}$ True Arithmetic

A <u>theory</u> Σ is a set of sentences (over $\mathcal{L}_A$) closed under logical consequence

- We can specify a theory by a subset of sentences that logically implies all sentences in Σ

Σ is <u>consistent</u> iff $\Phi_0 \neq \Sigma$ (iff $\forall A \in \Phi_0$, either A or $\neg A$ Not in Σ)

Σ is <u>complete</u> iff Σ is consistent and $\forall A$ either A or $\neg A$ is in Σ

Σ is <u>sound</u> iff $\Sigma \subseteq TA$

Let $\mathfrak{M}$ be a model/structure over $\mathcal{L}_A$

$$Th(\mathfrak{M}) = \{ A \in \Phi_0 \mid \mathfrak{M} \models A \}$$

$Th(\mathfrak{M})$ is <u>complete</u> (for all structures $\mathfrak{M}$)

<u>Note</u> $TA = Th(\mathbb{N})$ is complete, consistent, & sound

$$VALID = \{ A \in \Phi_0 \mid \models A \}$$ ← smallest theory

Let Σ be a theory

Σ is <u>axiomatizable</u> if there exists a set $\Gamma \subseteq \Sigma$

such that ① Γ is recursive

② $\Sigma = \{ A \in \Phi_0 \mid \Gamma \models A \}$

<u>Theorem</u> Σ is axiomatizable iff Σ is r.e.

(P. 76 of Notes)

Let Σ be a theory

Σ is axiomatizable if there exists a set $\Gamma \subseteq \Sigma$

such that ① Γ is recursive

② $\Sigma = \{ A \in \Phi_0 \mid \Gamma \models A \}$

Theorem Σ is axiomatizable iff Σ is r.e.

Proof $\Rightarrow$. Suppose Σ is axiomatizable, Γ recursive

Define $R(x,y)$ = true iff y encodes a Γ-LK proof of (the formula encoded by) x

R is recursive, so by previous *Theorem, Σ is r.e.

Let Σ be a theory

Σ is <u>axiomatizable</u> if there exists a set $\Gamma \subseteq \Sigma$

such that ① Γ is recursive

② $\Sigma = \{A \in \Phi_0 \mid \Gamma \models A\}$

<u>Theorem</u> Σ is axiomatizable iff Σ is r.e.

<u>Proof</u> $\Rightarrow$. Suppose Σ is axiomatizable, Γ recursive

Define $R(x,y)$ = true iff y encodes a Γ-LK proof of (the formula encoded by) x

R is recursive, so by previous *Theorem, Σ is r.e.

$\Leftarrow$ By *Theorem, Σ = range of total computable function f

$\therefore \quad \Sigma = \{f(0), f(1), f(2), \ldots\}$

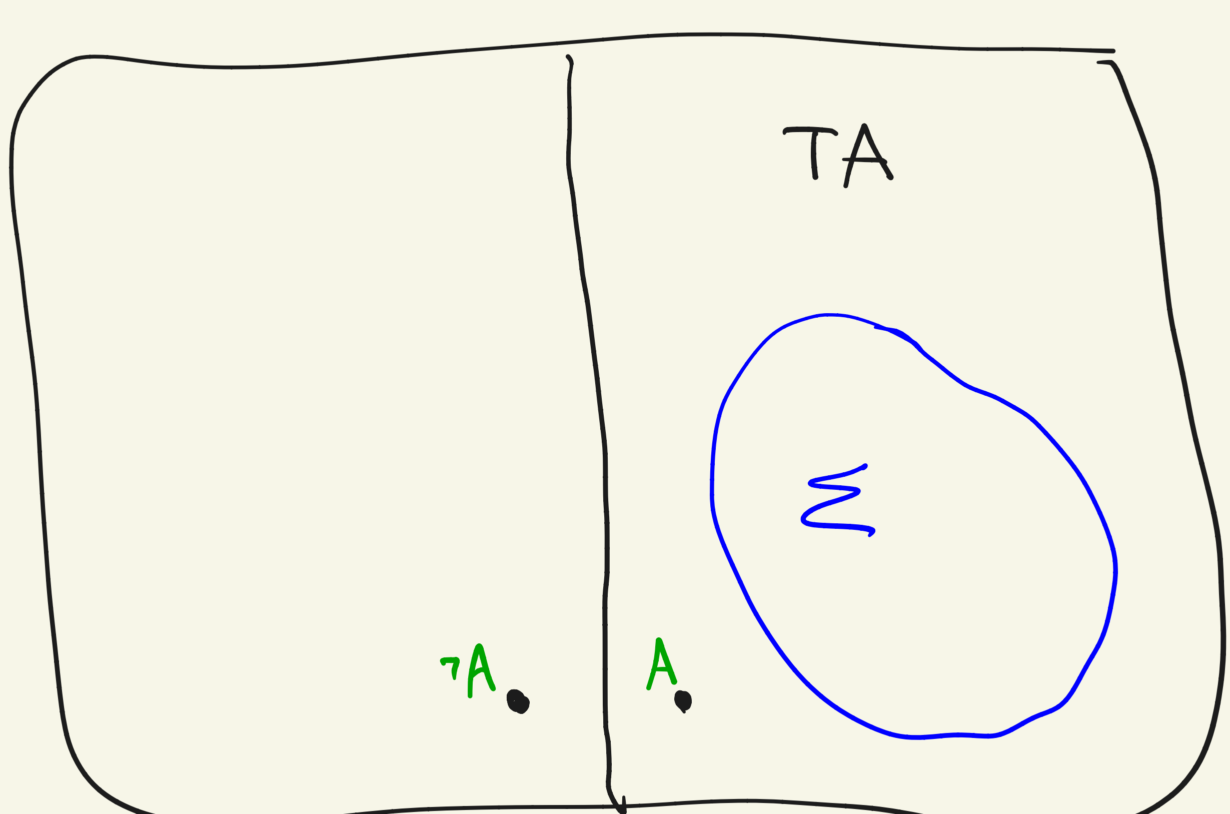
Incompleteness - Introduction

① TA is not r.e. (so by previous theorem, not axiomatizable)

First Incompleteness Theorem Every sound axiomatizable theory is incomplete

Φ_0:

all $\mathcal{L}_A$ sentences



Σ sound and axiomatizable $\Longrightarrow \exists A, \neg A \notin \Sigma$

Incompleteness - Introduction

1. TA is not r.e. (so by previous theorem, not axiomatizable)

   First Incompleteness Theorem Every sound axiomatizable theory is incomplete

2. Define PA - Peano arithmetic

   Sound, axiomatizable

   So by Tarski's Thm, PA is incomplete

3. Gödel's second Incompleteness Thm:

   A specific sentence asserting "PA is consistent" is not a theorem of PA

FIRST INCOMPLETENESS THEOREM

We define a predicate $\mathrm{Truth} \subseteq \mathbb{N}$

$$\mathrm{Truth} = \{ n \mid n \text{ encodes a sentence } \langle n \rangle \in \Phi_0 \text{ that is in } TA \}$$

We will show that Truth is not r.e.

FIRST INCOMPLETENESS THEOREM

We define a predicate $Truth \subseteq \mathbb{N}$

$$Truth = \{ m \mid m \text{ encodes a sentence } \langle m \rangle \in \Phi_0 \text{ that is in } TA \}$$

We will show that Truth is not r.e.:

Defn A predicate is arithmetical if it can be represented by a formula over $\mathcal{L}_A$

We'll show:

1. Every r.e. predicate/language is arithmetical
2. Truth is not arithmetical

$\therefore$ Truth is not r.e.

Since Truth is not r.e.,

there is NO r.e. TM that accepts exactly the sentences in TA

$\therefore$ TA is NOT axiomatizable

$\therefore$ Any sound, axiomatizable theory Σ is incomplete

(There is a sentence $A \in \Phi_0$ such that neither A or $\neg A$ are in Σ.)

FIRST INCOMPLETENESS THEOREM

We define a predicate $Truth \subseteq \mathbb{N}$

$$Truth = \{ m \mid m \text{ encodes a sentence } \langle m \rangle \in \Phi_0 \text{ that is in } TA \}$$

We will show that Truth is not r.e.:

Defn A predicate is arithmetical if it can be represented by a formula over $\mathcal{L}_A$

Show

1. Every r.e. predicate/language is arithmetical
2. Truth is not arithmetical

$\therefore$ Truth is not r.e.

Exists-Delta Theorem pp 68-71

Tarski Theorem pp. 73-74

① Every R.e. predicate is arithmetical

Definition Let $s_0 = 0$, $s_1 = s0$, $s_2 = ss0$, etc.

Let $R(x_1 \ldots x_n)$ be an n-ary relation $R \subseteq \mathbb{N}^n$

Let $A(x_1, \ldots, x_n)$ be an $\mathcal{L}_A$ formula, with free variables $x_1, \ldots, x_n$

$A(\vec{x})$ represents R iff $\forall \vec{a} \in \mathbb{N}^n \quad R(\vec{a}) \iff \mathbb{N} \models A(s_{a_1}\, s_{a_2} \ldots s_{a_n})$

Example $R \subseteq \mathbb{N} \quad R = \{ a \in \mathbb{N} \mid a \text{ is even} \}$

$$A : \exists y \, (y + y = x)$$

$3 \notin R$, and $\mathbb{N} \not\models A(ss0) = \exists y \, (y + y = sss0)$

$4 \in R$, and $\mathbb{N} \models A(sss0) = \exists y \, (y + y = ssss0) \qquad y = ss0$

① Every R.e. predicate is arithmetical

Definition Let $s_0 = 0$, $s_1 = s0$, $s_2 = ss0$, etc.

Let $R(x_1 \ldots x_n)$ be an n-ary relation $R \subseteq \mathbb{N}^n$

Let $A(x_1, \ldots, x_n)$ be an $\mathcal{L}_A$ formula, with free variables $x_1, \ldots, x_n$

$A(\vec{x})$ represents R iff $\forall \vec{a} \in \mathbb{N}^n \quad R(\vec{a}) \Longleftrightarrow \mathbb{N} \models A(s_{a_1}, s_{a_2} \ldots s_{a_n})$

R is arithmetical iff there is a formula $A \in \mathcal{L}_A$ that represents R

Exists-Delta-Theorem every r.e. relation is arithmetical. In fact every r.e. relation is represented by a $\exists\Delta_0$ $\mathcal{L}_A$-formula.

$\exists\Delta_0$ Formulas

$t_1 \le t_2$ stands for $\exists z(t_1 + z = t_2)$

$$\left.\begin{array}{lll} \exists x \le t\ A & \text{stands for} & \exists x\,(x \le t \wedge A) \\ \forall x \le t\ A & \text{stands for} & \forall x\,(x \le t \supset A) \end{array}\right\} \text{Bounded Quantifiers}$$

Definition A formula is a Δ_0-formula if it has the form

$$\underbrace{\forall x_1 \le t_1\ \exists x_2 \le t_2\ \forall x_3 \le t_3 \ldots \exists x_k \le t_k}_{\text{Bounded Quantifiers}}\ \underbrace{A(x_1 \ldots x_k\ \vec{y})}_{\text{No quantifiers}}$$

Definition A relation $R(\vec{x})$ is a Δ_0-relation iff some Δ_0-formula represents it

$\exists\Delta_0$ Formulas

<u>Example</u> $Prime = \{x \in \mathbb{N} \mid x \text{ is prime}\}$ is a Δ_0-relation, represented by the following Δ_0-formula:

$$S0 < x \wedge \forall z \leq x\ \forall y \leq x\ (x = z \cdot y \supset (z = 1 \vee z = x))$$

$\exists\Delta_0$ Formulas

$t_1 \leq t_2$ stands for $\exists w\,(t_1 + w = t_2)$

$\exists z \leq t\ A$ stands for $\exists z\,(z \leq t \wedge A)$

$\forall z \leq t\ A$ stands for $\forall z\,(z \leq t \supset A)$

$\Big\}$ Bounded Quantifiers

Definition A formula is a Δ_0-formula if it has the form $\forall z_1 \leq t_1\ \exists z_2 \leq t_2\ \forall z_3 \leq t_3 \ldots \exists z_k \leq t_k\ A(z_1 \ldots z_k, \vec{x})$

Definition A $\exists\Delta_0$ formula has the form $\exists y \underbrace{B}_{\Delta_0\text{ formula}}$

Definition A relation $R(\vec{x})$ is a Δ_0-relation iff some Δ_0-formula represents it

Definition $R(\vec{x})$ is a $\exists\Delta_0$-relation iff some $\exists\Delta_0$-formula represents it

$\exists\Delta_0$ Formulas

<u>Lemma</u> Every Δ_0 relation is recursive

<u>Lemma</u> Every $\exists\Delta_0$ relation is r.e.

<u>$\exists\Delta_0$ (Exists-Delta) Theorem</u> every r.e. relation is represented by a $\exists\Delta_0$ formula

$\exists\Delta_0$ Theorem

Main Lemma Let $f : \mathbb{N}^n \to \mathbb{N}$ be a total computable function.

Let $R_f = \{ (\vec{x}, y) \in \mathbb{N}^{n+1} \mid f(\vec{x}) = y \}$ also called graph(f)

Then R_f is a $\exists\Delta_0$-relation.

Main Lemma Let $f: \mathbb{N}^n \to \mathbb{N}$ be total, computable

Then $R_f = \{(\vec{x}, y) \mid f(\vec{x}) = y\}$ is a $\exists\Delta_0$ relation

---

Proof of $\exists\Delta_0$ Theorem from Main Lemma

Let $R(\vec{x})$ be an r.e. relation

Then $R(\vec{x}) = \exists y\, S(\vec{x}, y)$ where S is recursive

Since S is recursive, $f_S(\vec{x}, y) = \begin{cases} 1 & \text{if } (\vec{x}, y) \in S \\ 0 & \text{otherwise} \end{cases}$

is total computable

By main lemma R_{f_S} is represented by a $\exists\Delta_0$ relation

So $R(\vec{x}) = \exists y \underbrace{\exists z\, B}_{R_{f_S}}$ is represented by a $\exists\Delta_0$ relation

Proof of Main Lemma (see pp 70-71)

Main idea: is a way of representing sequences of numbers by numbers using $\exists\Delta_0$ formulas

Note: Prime power decomposition not useful here since we only have $s, +, \cdot$
(ie. represent (a_1, a_2, a_3, a_4) by $2^{a_1} \cdot 3^{a_2} \cdot 5^{a_3} \cdot 7^{a_4}$)

Definition β-function

$$\beta(c, d, i) = rm(c, d(i+1)+1)$$

where $rm(x, y) = x \bmod y$

Proof of Main Lemma (see pp 70-71)

Definition β-function

$$\beta(c, d, i) = rm(c, d(i+1)+1) \quad \text{where } rm(x, y) = x \bmod y$$

Lemma 0. $\forall n, r_0, r_1, \ldots, r_n \ \exists c, d$ such that

$$\beta(c, d, i) = r_i \quad \forall i, \ 0 \leq i \leq n$$

So the pair (c, d) represents the sequence $r_0 r_1, \ldots r_n$ using β

Proof of Main Lemma (see pp 70-71)

Definition β-function

$$\beta(c, d, i) = rm(c, d(i+1)+1) \quad \text{where } rm(x, y) = x \bmod y$$

Lemma 0. $\forall n, r_0, r_1, \dots, r_n \ \exists c, d$ such that

$$\beta(c, d, i) = r_i \quad \forall i, \ 0 \le i \le n$$

CRT (Chinese Remainder Theorem)

Let $r_0, \dots, r_n$, $m_0, \dots, m_n$ be such that $0 \le r_i \le m_i \ \forall i, \ 0 \le i \le n$ and $\gcd(m_i, m_j) = 1 \ \forall i, j$

Then $\exists r$ such that $rm(r, m_i) = r_i \ \forall i, \ 0 \le i \le n$

Proof of Main Lemma (see pp 70-71)

$\boxed{\textbf{Lemma } \Delta \quad \forall n, r_0, r_1, \ldots, r_n \ \exists c, d \text{ such that } \beta(c,d,i) = r_i \ \forall i,\ 0 \le i \le n}$

$\beta(c,d,i) = rm(c, d(i+1)+1)$

where $rm(x,y) = x \bmod y$

Chinese Remainder Theorem

Let $r_0, \ldots, r_n$, $m_0, \ldots, m_n$ be such that $0 \le r_i \le m_i$ and $\gcd(m_i, m_j) = 1$. Then $\exists r$ $rm(r, m_i) = r_i$ $\forall i$

Proof of Lemma Δ

Let $d = (n + r_0 + \ldots + r_n + 1)!$

Let $m_i = d(i+1)+1$

Claim $\forall i,j$ $\gcd(m_i\, m_j) = 1$ (see Notes)

By CRT $\exists r = c$ so that $\beta(c,d,i) = rm(c, m_i) = r_i$ $\forall i \in [n]$

Proof of Main Lemma (see pp 70-71)

Lemma 0 $\forall n, r_0, r_1, \ldots, r_n \ \exists c, d$ such that

$$\beta(c,d,i) = r_i \quad \forall i, \ 0 \le i \le n$$

Lemma 1 R_β is a Δ_0 relation

Pf: $$y = \beta(c,d,i) \iff [\exists q \le c\, (c = q(d(i+1)+1)+y) \wedge y < d(i+1)+1]$$

Lemma 2 If $R(\vec{x},y)$ is a $\exists\Delta_0$ relation, R_β is a $\exists\Delta_0$ relation

then $S(\vec{x}) = \exists y\, (R_\beta(\vec{x},y) \wedge R(\vec{x},y))$ is a $\exists\Delta_0$ relation

Proof of Main Lemma (see pp 70-71)

Let $f : \mathbb{N} \to \mathbb{N}$ be unary, total computable function, & let M_f be TM computing f

$R(\vec{x}, y)$ will be a $\exists\Delta_0$ relation saying:

$\exists m, c, d$ such that

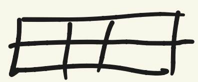
(1) c, d describe the tableaux given by $r_1 \dots r_m \dots r_{m^2}$ given by β function

(2) $r_1 \dots r_m$ encode start config of M_f on x

(3) Last m numbers $r_{(m-1)m} \dots r_{m^2}$ encode last config, containing y in first cells then B, and state is q_2

(4) For all other configs, state is not q_2.

(5) all 2×3 local cells are consistent with transition function of M_f



Recap: we wanted to prove

<u>$\exists\Delta_0$ (Exists-Delta) Theorem</u> every r.e. relation is represented by a $\exists\Delta_0$ formula

which followed by <u>Main Lemma</u>:

f total, computable $\Rightarrow R_f$ is a $\exists\Delta_0$ relation

FIRST INCOMPLETENESS THEOREM

We define a predicate $\text{Truth} \subseteq \mathbb{N}$

$$\text{Truth} = \{ m \mid m \text{ encodes a sentence } \langle m \rangle \in \Phi_0 \text{ that is in TA} \}$$

We will show that Truth is not r.e.:

Defn A predicate is arithmetical if it can be represented by a formula over $\mathcal{L}_A$

1. Every r.e. predicate/language is arithmetical DONE!

Exists-Delta Theorem pp 68-71

2. Truth is not arithmetical

Tarski Theorem pp. 73-74

$\therefore$ Truth is not r.e.

Tarski Theorem

Define the predicate $\text{Truth} \subseteq \mathbb{N}$

$$\text{Truth} = \{ n \mid n \text{ encodes a sentence } \langle n \rangle \in TA \}$$

Then Truth is not arithmetical

Tarski Theorem

Define the predicate $\mathrm{Truth} \subseteq \mathbb{N}$

$$\mathrm{Truth} = \{ n \mid n \text{ encodes a sentence } \langle n \rangle \in TA \}$$

Then Truth is not arithmetical

High Level idea:

Formulate a sentence "I am false"

which is self-contradictory

Pf of Tarski's Thm

Let $sub(m,n) = \begin{cases} 0 \text{ if } m \text{ is not a legal encoding of a formula} \\ \text{otherwise say } m \text{ encodes the formula} \end{cases}$

$A(x)$ with free variable x.

Then $sub(m,n) = m'$ where m' encodes $A(s_n)$

Let $d(n) = sub(n,n)$

$$\left\{ \begin{array}{l} d(n) = 0 \text{ if } n \text{ not a legal encoding.} \\ \text{ow say } n \text{ encodes } A(x). \\ \text{then } d(n) = n' \text{ where } n' \text{ encodes } A(s_n) \end{array} \right\}$$

clearly sub, d are both computable

Proof of Tarski's Thm

Suppose that Truth is arithmetical.

Then define $R(x) = \neg \mathrm{Truth}(d(x))$

Since d, Truth both arithmetical, so is R

Let $\widetilde{R}(x)$ represent $R(x)$, and let e be the encoding of $\widetilde{R}(x)$

Let $d(e) = e'$ so e' encodes $\widetilde{R}(s_e)$ encodes "I am false"

Then

$\widetilde{R}(s_e) \in TA \iff \neg \mathrm{Truth}(d(e'))$ since $\widetilde{R}$ represents R

$\iff \neg \widetilde{R}(s_e) \in TA$ by defn of truth

$\iff \widetilde{R}(s_e) \notin TA$ TA contains exactly one of $A, \neg A$

↯ this is a contradiction. $\therefore$ Truth is not arithmetical

FIRST INCOMPLETENESS THEOREM

FINALLY WE HAVE PROVEN:

(1) Every r.e. predicate/language is arithmetical

Exists-Delta Theorem pp. 68-71

(2) Truth is not arithmetical

Tarski Theorem 73-74

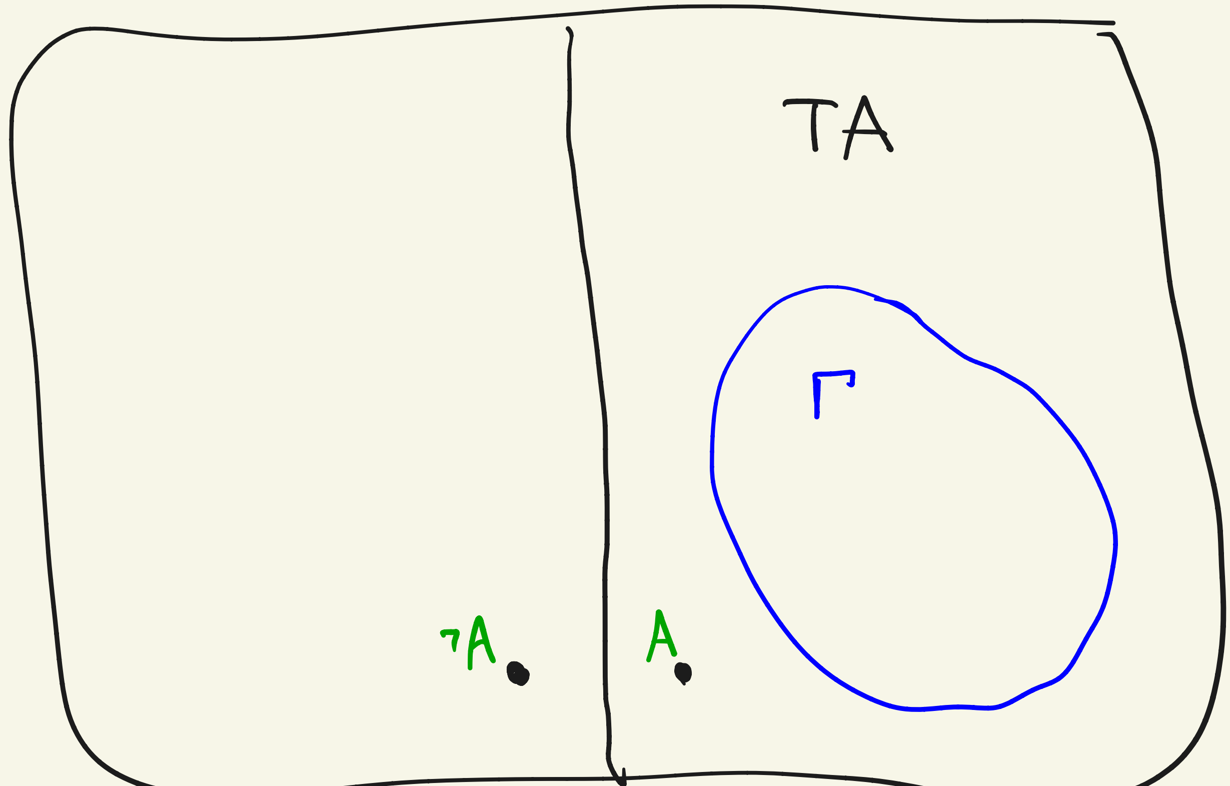
∴ Truth is not r.e.

Truth not r.e. ⇒ TA Not axiomatizable

∴ Any SOUND, axiomatizable theory is incomplete

Φ_0:

all $\mathcal{L}_A$ sentences



Γ sound and axiomatizable $\Rightarrow \exists A, \neg A \notin \Gamma$

2nd INCOMPLETENESS THEOREM

- We will define PA (Peano Arithmetic), an axiomatizable sound theory.
- Most of number theory provable in PA
- We will see that PA cannot prove its own consistency (2nd Incompleteness Thm)