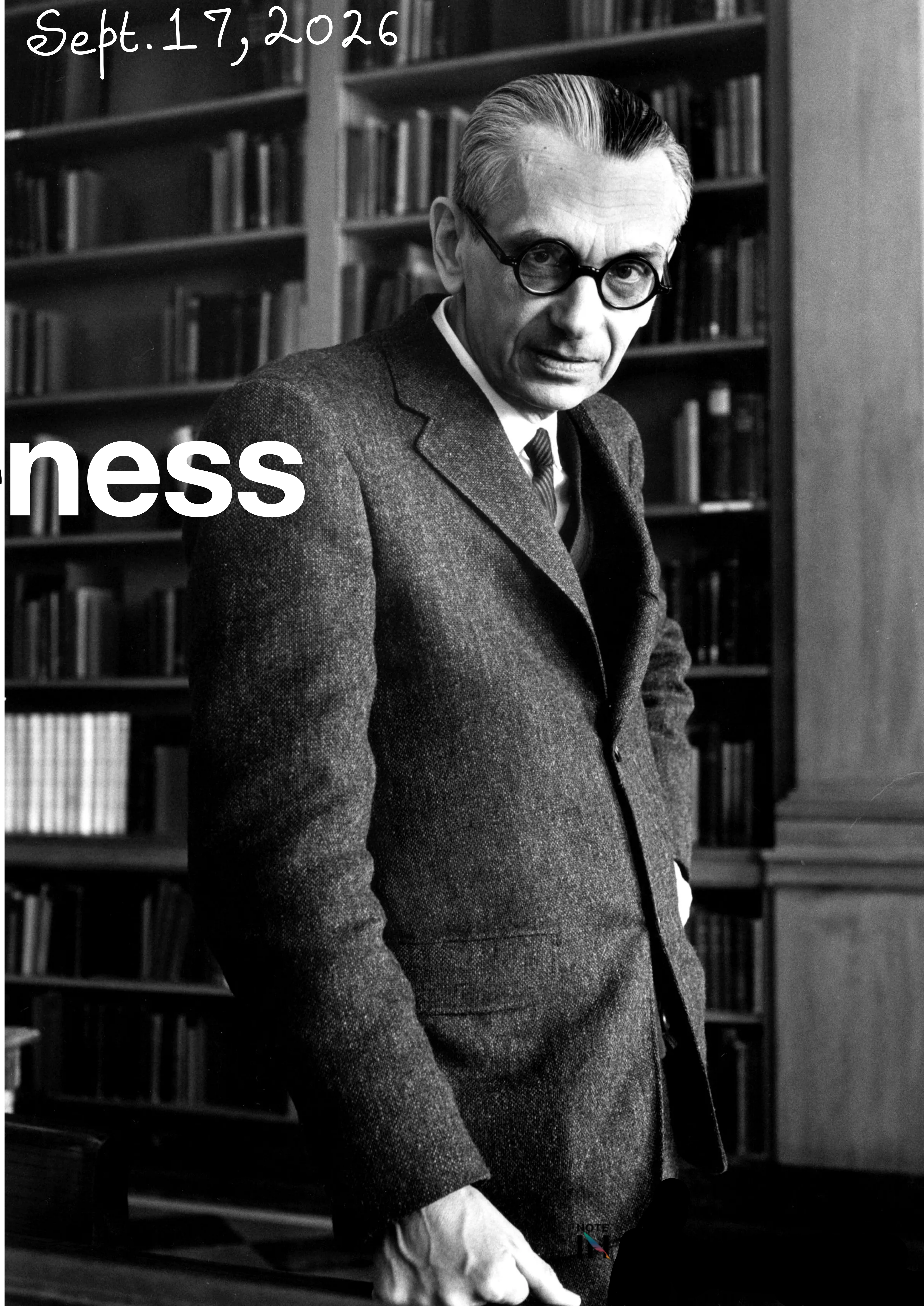
Sept. 17, 2026



Gödel’s Incompleteness Theorems

A formal study of introspection

Saptarshi Sahoo

Primitive Recursive(p.r) Functions

every

A collection of functions of form $f : \mathbb{N}^n \to \mathbb{N}$ for ~~some~~ $n \in \mathbb{N}$

- Every $Z : \mathbb{N}^n \to \mathbb{N}$ which are identically zero is in p.r., $\forall n \in \mathbb{N}$
- For each $1 \leq k \leq n$, the function $P_k^n : \mathbb{N}^n \to \mathbb{N}$ defined by $(x_1, x_2, x_3, \cdots, x_n) \mapsto x_k$ is in p.r.
- $S : \mathbb{N} \to \mathbb{N}$ defined as $n \mapsto n + 1$is in p.r.
- If $f : \mathbb{N}^n \to \mathbb{N}$ is in p.r and for each $1 \leq k \leq n$, $g_k : \mathbb{N}^m \to \mathbb{N}$ is in p.r, then $h : \mathbb{N}^m \to \mathbb{N}$ is in p.r where h is defined by

$$h(x_1, x_2, \cdots, x_m) = f(g_1(x_1, x_2, \cdots, x_m), g_2(x_1, x_2, \cdots, x_m), \cdots, g_n(x_1, x_2, \cdots, x_m))$$

Notation: $h(x_1, x_2, \cdots, x_m) = f \circ (g_1, g_2, g_3, \cdots, g_n)(x_1, x_2, \cdots, x_m)$

- If $g : \mathbb{N}^{n+2} \to \mathbb{N}$and $h : \mathbb{N}^{n-1} \to \mathbb{N}$ are both in p.r, then $f : \mathbb{N}^n \to \mathbb{N}$ is in p.r where,

Primitive Recursive(p.r) Functions

- $f(x+1, x_1, \cdots, x_{n-1}) = \begin{cases} h(x_1, x_2, \cdots, x_{n-1}) & \text{If the first argument is zero} \\ g(x, f(x, x_1, x_2, \cdots, x), x_1, x_2, \cdots, x_{n-1}) & \text{If } x \geq 1 \end{cases}$

If you want more details:

(1) Chap. 14 of Peter Smith's book

(2) Wikipedia page on Primitive recursive function

Primitive Recursive(p.r) Functions

Some examples:

1. $Add(m, n) : \mathbb{N}^2 \to \mathbb{N}$

$$Add(0,n) = P_1^1(n)$$

$$A(S\ m, n) = S \circ P_2^4(m, Add(m, n), m, n) \ = S \circ Add(m, n)$$

2. $Mult(m, n) : \mathbb{N}^2 \to \mathbb{N}$

$$Mult(0,n) = Z(n)$$

$$Mult(S\ m, n) = Add \circ (P_2^4, P_4^4)(m, Mult(m, n), m, n) = Add \circ (Mult(m, n), n)$$

Primitive Recursive(p.r) Functions

Another way to look at it

- Take the multiplication example from the previous slide. Suppose we want to calculate $Mult(3,2)$. We can go as follows:

- $Mult(0,2) = 0$, $Mult(1,2) = Add(Mult(0,2),2) = 2$, $Mult(2,2) = Add(Mult(1,2),2) = 4$,$Mult(3,2) = Add(Mult(2,2),2) = 6$.

- Couple of things : (1) Finite List. (2) Does it look like something familiar to something that you had probably done in a programming course?

Wild $\beta(\cdot, \cdot, \cdot, \cdot)$ function will appear in future!

Δ_0 sentences (Ref: Ch. 11)

Nice Sentences

A little bit of notation:

$x \leq y : \exists z(x + z = y)$

$\forall x \leq y(\varphi(x)) : \forall x(x \leq y \to \varphi(x))$

$\exists x \leq y(\varphi(x)) \ : \ \exists x(x \leq y \wedge \varphi(x))$

- If σ and τ are terms , then σ = τ and σ ≤ τ are in Δ_0.
- If φ and ψ are in Δ_0, so are their boolean combination and their negation.
- If φ is in Δ_0, then $\forall x \leq \tau\ \varphi$ and $\exists x \leq \tau\ \varphi$ are in Δ_0. Where τ is a term and x is free in φ.
- Nothing else is.

Fact: PA is decides all Δ_0 sentences!

Σ_1-sentences

Almost nice Sentences

- Any Δ_0 formula is a Σ_1 formula .
- If φ and ψ are in Σ_1, so are $\varphi \wedge \psi$ and $\varphi \vee \psi$
- If φ is in Σ_1, then $\forall x \leq \tau \ \varphi$ and $\exists x \leq \tau \ \varphi$ are in Σ_1. Where τ is a term and x is free in φ.
- If φ is in Σ_1, so is $\exists x \ \varphi$ where x is any variable free in φ.
- Nothing else is Σ_1.

First Order Logic (Reference: Ch. 5, 16)

Function *capturing* in a theory $\boldsymbol{T}$ and function *expressing* in a Language

- Let, $f : \mathbb{N} \to \mathbb{N}$ be a function from natural numbers to natural numbers.

- We say a theory T captures $f : \mathbb{N} \to \mathbb{N}$ by a formula $\varphi(\,\cdot\,,\,\cdot\,)$ if:

  $\forall m, n \in \mathbb{N}$

  1. $T \vdash \varphi(\overline{m}, \overline{n})$, if $f(m) = n$.

  2. $T \vdash \exists! y \varphi(\overline{m}, y)$

  → Syntactic notion

- We say function is expressible in $\mathscr{L}$ if:

  A. $\mathbb{N} \vDash \varphi(\overline{m}, \overline{n})$, whenever $f(m) = n$

  B. $\mathbb{N} \vDash \neg\varphi(\overline{m}, \overline{n})$, whenever $f(m) \neq n$

  → Semantic Notion

Nikhil's doubt (Pardon me! I forgot which Nikhil ☹)

Defn. 1 for expressible function in $\mathcal{L}_{PA}$: (Same as above)

A. If $f(m)=n$, then $\mathbb{N} \models \varphi(\bar{m},\bar{n})$

B. If $f(m)\neq n$, then $\mathbb{N} \models \neg\varphi(\bar{m},\bar{n})$

Defn. 2:

If $f(m)=n$ then,

(A) $\mathbb{N} \models \varphi(\bar{m},\bar{n})$

(B') $\mathbb{N} \models \exists! y\, \varphi(\bar{m},y)$

Claim: Both are equivalent definition

Proof: Exercise. Just follow intuition.

First Order Logic

Function *capturing* in a theory T and function *expressing* in a Language

- Let, $f : \mathbb{N} \to \mathbb{N}$ be a function from natural numbers to natural numbers.
- We say a theory T captures $f : \mathbb{N} \to \mathbb{N}$ by a formula $\varphi(\cdot, \cdot)$ if:

1. $T \vdash \varphi(\overline{m}, \overline{n})$, if $f(m) = n$.
2. $T \vdash \exists! y \varphi(\overline{m}, y)$

Fact 1: PA can prove true Σ_1 sentences.
Fact 2: p.r functions are expressed by Σ_1
Fact 3: Hence PA can capture p.r functions by Σ_1 sentences

- We say function is expressible in $\mathscr{L}$ if:

A. $\mathbb{N} \vDash \varphi(\overline{m}, \overline{n})$, whenever $f(m) = n$

B. $\mathbb{N} \vDash \neg\varphi(\overline{m}, \overline{n})$, whenever $f(m) \neq n$

Pulling the meta-theory inside formal theory

Some tools...

$\neg A := A \to \bot$

$A \vee B := \neg A \to B$

$A \wedge B := \neg(A \to \neg B)$

Prime numbers:2, 3, 5, 7, 11, 13, 17
, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59,
61, 67, 71, 73, 79, 83, 89, 97, 101, 103,
107, 109, 113, 127, 131, 137, 139...

- Symbols we have:

$$\bot\,,\ \to\,,\ \forall,\ =\,,\ 0,\ S,\ +\,,\ \times\,,\ (,\ )$$

- We can put Gödel numbering as $\overline{1}$, $\overline{3}$, $\overline{5}$, $\overline{7}$, $\overline{9}$, $\overline{11}$, $\overline{13}$, $\overline{15}$, $\overline{17}$ and $\overline{19}$. Rest of the Odd numbers are for variables. Something like this: $\ulcorner v_i \urcorner := 2 \times i + 21$

- Then for a formula $\ulcorner \forall v_0 (v_1 = S00 \times v_0 \to \bot\,) \to \bot\, \urcorner =$

$$2^{\ulcorner\forall\urcorner} \times 3^{\ulcorner v_0\urcorner} \times 5^{\ulcorner(\urcorner} \times 7^{\ulcorner v_1\urcorner} \times 11^{\ulcorner=\urcorner} \times 13^{\ulcorner S\urcorner} \times 17^{\ulcorner0\urcorner} \times 19^{\ulcorner0\urcorner} \times 23^{\ulcorner v_0\urcorner} \times 29^{\ulcorner\to\urcorner} \times 31^{\ulcorner\bot\urcorner} \times 37^{\ulcorner)\urcorner} \times 41^{\ulcorner\to\urcorner} \times 43^{\ulcorner\bot\urcorner}$$

$$= 2^5 \times 3^{21} \times 5^{17} \times 7^{23} \times 11^7 \times 13^{11} \times 17^9 \times 19^9 \times 23^{21} \times 29^3 \times 31^1 \times 37^{19} \times 41^3 \times 43^1$$

Revisiting to p.r. functions (Ref. Ch-14)

For loops in disguise… (P.pp-108)

- A relation $P(\bar{x})$ is a p.r. relation, if the characteristic function is a p.r. function.

- If $f(\bar{x})$ is an n-place p.r. function, then the corresponding relation expressed by $f(\bar{x}) = y$ is an $n + 1$-place p.r. relation.

- Any truth-functional combination of p.r. relations is p.r.

- Any relation defined from a p.r. property or relation by bounded quantifications is also p.r.

- If P is a p.r. relation, then the function $f(n) = (\mu x \leq n)P(x)$ is p.r And generalizing, suppose that $g(n)$ is a p.r. function; then $f'(n) = (\mu x \leq g(n))P(x)$ is also p.r.

- Any function defined by cases from other p.r. functions is also p.r.

P.S. : Use different LLM tools when needed, but don't be lazy; dig deep into the concepts.

p.r. are for loops in disguise. Say $f: \mathbb{N}^2 \longrightarrow \mathbb{N}$ is defined recursively as following

$$f(0, m) = g(m)$$
$$f(n, m) = h(n-1, f(n-1, m), m).$$

Equivalent for loop:

```
def func_f(n, m):
f := func_g(m)
for i := 0 to n:
    f := func_h(i, f, m)
return f
```

where, func_g(·) represents $g: \mathbb{N} \to \mathbb{N}$ from above.
func_h(·,·,·) represents $h: \mathbb{N}^3 \to \mathbb{N}$ from above.

<u>Exercise:</u>

(a) Show monus function; $m \dot{-} n = |m - n|$ is Primitive Recursive.

(b) Try to make yourself comfortable with the fact that can do basic if-then-else in p.r. function.