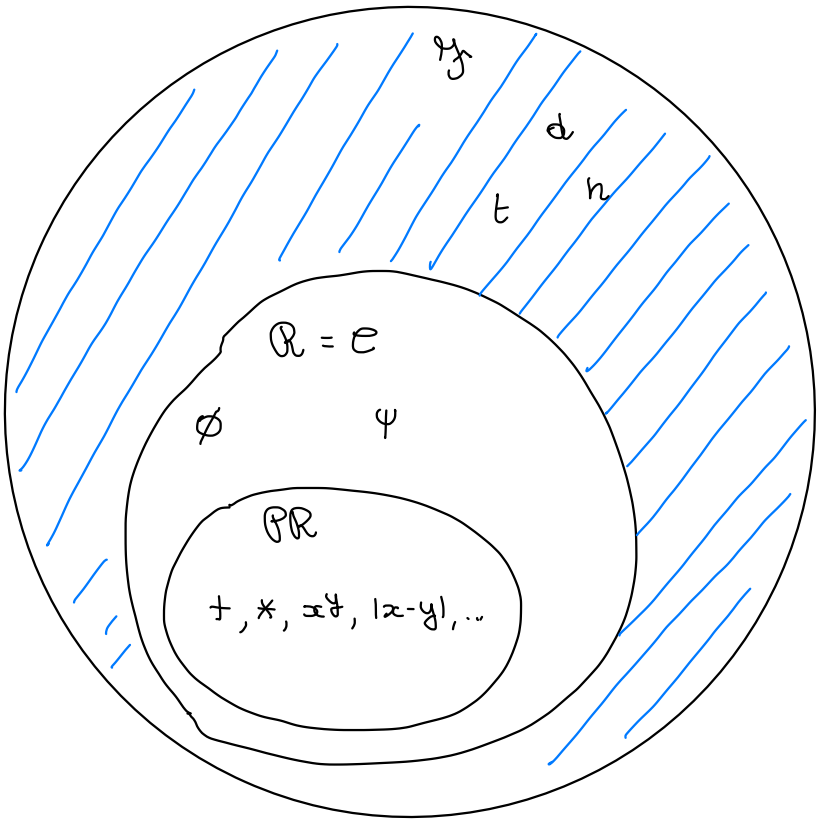
COMPUTABILITY (17/11/2026)

* Recursive and Recursively Enumerable sets



$$d(x) = \begin{cases} \varphi_x(x)+1 & \text{if } \varphi_x(x)\downarrow \\ 0 & \text{otherwise} \end{cases}$$

$$h(x) = \begin{cases} 1 & \text{if } \varphi_x(x)\downarrow \\ 0 & \text{otherwise} \end{cases}$$

$$t(x) = \begin{cases} 1 & \text{if } W_x = \mathbb{N} \\ 0 & \text{otherwise} \end{cases}$$

given $X \subseteq \mathbb{N}$ ↑ set of programs (property)

"$x \in X$?"

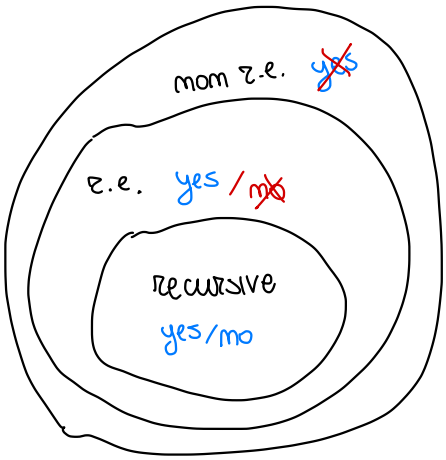
$X = \{ x \mid \varphi_x = fact \}$

$X = \{ x \mid P_x$ has linear complexity $\}$

$X = \{ x \mid P_x$ does not access to $R_y \}$

$X = \{ x \mid P_x$ executes each instruction for some input $\}$

⋮



decidable properties / recursive sets

answer: yes/no

semi-decidable properties / recursively enumerable sets (r.e.)

answer yes, ~~no~~

* Recursive sets

A set $A \subseteq \mathbb{N}$ is <u>recursive</u> if the characteristic function

$$\chi_A : \mathbb{N} \to \mathbb{N}$$

$$\chi_A(x) = \begin{cases} 1 & \text{if } x \in A \\ 0 & \text{if } x \notin A \end{cases}$$

is computable

($\leftrightarrow$ predicate "$x \in A$" decidable)

<u>Examples</u> :

$\mathbb{N}$	recursive	$\chi_{\mathbb{N}}(x) = 1$	$\forall x \in \mathbb{N}$	computable
$\emptyset$	"	$\chi_{\emptyset}(x) = 0$	$\forall x \in \mathbb{N}$	"
$\mathbb{P}$	"	$\chi_{\mathbb{P}}(x) = \overline{sg}(rm(2,x))$		

OBSERVATION : All finite sets $A \subseteq \mathbb{N}$ are recursive

<u>proof</u>

let $A = \{x_0, x_1, \dots, x_m\}$

$$\chi_A(x) = \overline{sg}\left(\prod_{i=0}^{m} |x - x_i|\right)$$ computable

□

OBSERVATION : if $A, B \subseteq \mathbb{N}$ recursive sets then

(i) $\overline{A} = \{x \in \mathbb{N} \mid x \notin A\} = \mathbb{N} \setminus A$

(ii) $A \cap B$ are recursive

(iii) $A \cup B$

<u>proof</u>

same as for decidable predicates

e.g. (iii)

$$\chi_{A \cup B}(x) = sg\underbrace{(\chi_A(x) + \chi_B(x))}_{\begin{array}{l} >0 \text{ if } x \in A \text{ or } x \in B \\ 0 \text{ otherwise} \end{array}}$$ computable.

* Halting Set

$$K = \{ x \in \mathbb{N} \mid \varphi_x(x) \downarrow \}$$
$$= \{ x \in \mathbb{N} \mid x \in W_x \}$$

NOT RECURSIVE

$$\chi_K(x) = \begin{cases} 1 & \text{if } \varphi_x(x) \downarrow \\ 0 & \text{otherwise} \end{cases}$$

not computable

* REDUCTION

problems Q and B

Q reduces to B — every instance of Q can be transformed easily into an instance of B (equivalent)

Def: Given $A, B \subseteq \mathbb{N}$

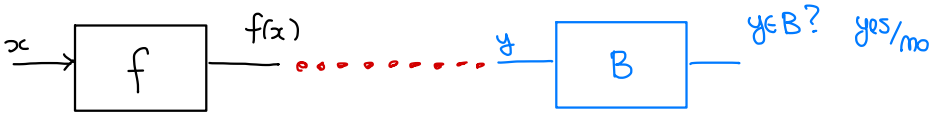
we say that the problem "$x \in A$" reduces to the problem "$x \in B$"

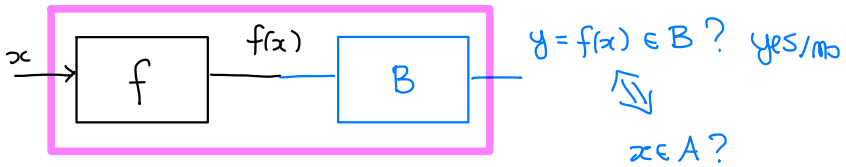
A reduces to B (in short) and write $A \leq_m B$

if there is a total computable function $f: \mathbb{N} \to \mathbb{N}$ s.t.

$\forall x \in \mathbb{N}$

$$x \in A \quad \text{iff} \quad f(x) \in B$$





OBSERVATION: Let $A, B \subseteq \mathbb{N}$ $\quad A \leq_m B$

(i) if B is recursive then A is recursive

(ii) if A is not recursive then B is not recursive

proof

(i) let B be recursive, i.e the function

$$\chi_B(x) = \begin{cases} 1 & \text{if } x \in B \\ 0 & \text{if } x \notin B \end{cases}$$ computable

let $A \leq_m B$ and let $f: \mathbb{N} \to \mathbb{N}$ the total computable reduction function $\forall x \in \mathbb{N}$

$$x \in A \iff f(x) \in B$$

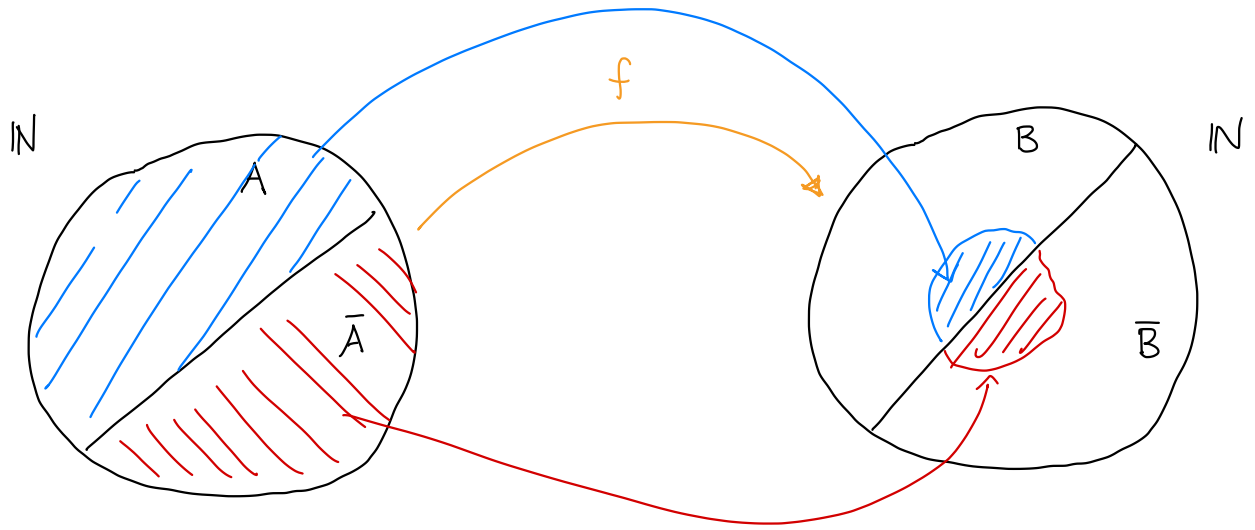
Then

$$\chi_A(x) = \begin{cases} 1 & \text{if } x \in A \quad (f(x) \in B) \\ 0 & \text{if } x \notin A \quad (f(x) \notin B) \end{cases} = \chi_B(f(x))$$

computable by composition

$\Rightarrow$ A recursive.

(ii) contronominal of (i).

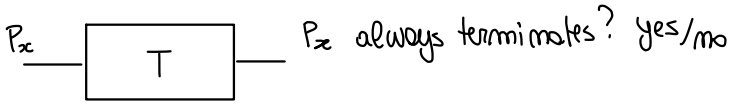


EXAMPLE $K = \{x \in \mathbb{N} \mid \varphi_x(x)\downarrow\} = \{x \in \mathbb{N} \mid x \in W_x\}$ not recursive

$T = \{x \in \mathbb{N} \mid \varphi_x \text{ total}\} = \{x \in \mathbb{N} \mid W_x = \mathbb{N}\}$

$K \leq_m T$

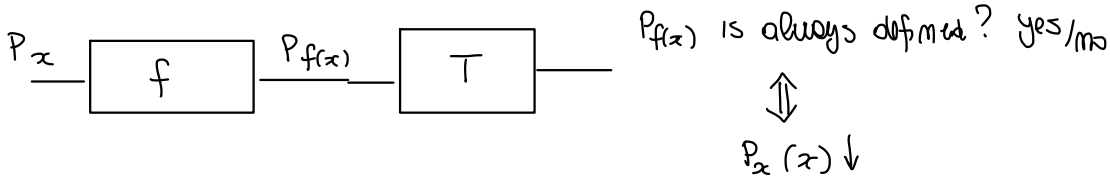
assume that T is recursive



given P_x we construct $P_{f(x)}$ s.t.

$P_x(x)\downarrow$ iff $P_{f(x)}$ is always terminating

then we can connect



The idea for defining f




```
def P_f(x)(y):
    return P_x(x)
```

if $P_x(x)\downarrow$ then $P_{f(x)}$ total

if $P_x(x)\uparrow$ then $P_{f(x)}$ not total

Formally

$g(x,y) = \varphi_x(x) = \psi_U(x,x)$ computable

By the smn theorem there is $f: \mathbb{N} \to \mathbb{N}$ total computable s.t. $\forall x,y$

$\varphi_{f(x)}(y) = g(x,y) = \varphi_x(x)$

We claim that f is the reduction function for $K \leq_m T$

i.e.

$\forall x \in \mathbb{N}$ $\quad x \in K$ iff $f(x) \in T$

* if $x \in K$ then $f(x) \in T$

if $x \in K$ then $\varphi_x(x)\downarrow$ and hence $\varphi_{f(x)}(y) = g(x,y) = \varphi_x(x)\downarrow \quad \forall y$

i.e. $\varphi_{f(x)}$ total, hence $f(x) \in T$

* if $x \notin K$ then $f(x) \notin T$

if $x \notin K$ then $\varphi_x(x)\uparrow$ and thus $\varphi_{f(x)}(y) = \varphi_x(x)\uparrow \quad \forall y$

i.e. $\varphi_{f(x)}$ not total and thus $f(x) \notin T$

This shows f is a reduction function for $K \leq_m T$

hence, since K not recursive, T is not recursive.

EXAMPLE (input problem)

Let $m \in \mathbb{N}$ fixed. Consider $A_m = \{ x \in \mathbb{N} \mid \varphi_x(m)\downarrow \}$

$K \leq_m A_m$

P_x defined on m iff $P_x(x)\downarrow$

```
def P_f(x)(y):
    return P_x(x)
```

- if $P_x(x)\downarrow$ then $P_{f(x)}(y)\downarrow \ \forall y$
  hence in particular $\varphi_{f(x)}(m)\downarrow$
- if $P_x(x)\uparrow$ then $P_{f(x)}(y)\uparrow \ \forall y$
  hence in particular $\varphi_{f(x)}(m)\uparrow$

Define $g : \mathbb{N}^2 \to \mathbb{N}$

$$g(x,y) = \varphi_x(x) = \psi_U(x,x)$$

computable, hence by smn theorem there is $f : \mathbb{N} \to \mathbb{N}$ s.t. $\forall x,y$

$$\varphi_{f(x)}(y) = g(x,y) = \varphi_x(x)$$

f is the reduction function $K \leq A_m$

* if $x \in K$ then $f(x) \in A_m$

if $x \in K$ then $\forall y \quad \varphi_{f(x)}(y) = g(x,y) = \varphi_x(x)\downarrow$, in particular

$\varphi_{f(x)}(m)\downarrow$ and thus $f(x) \in A_m$

* if $x \notin K$ then $f(x) \notin A_m$

if $x \notin K$ then $\varphi_{f(x)}(y) = g(x,y) = \varphi_x(x) \uparrow \quad \forall y$

in particular $\varphi_{f(x)}(m) \uparrow$ and thus $f(x) \notin A_m$

hence $K \leq_m A_m$ and thus A_m not recursive

* EXERCISE : Show that $A_m \leq_m K$

* EXERCISE : Show that $T \not\leq_m K$

* EXERCISE : Output problem

Let $m \in \mathbb{N}$ fixed. Consider $B_m = \{ x \in \mathbb{N} \mid m \notin E_x \}$

and show $B_m \leq_m K$

$K \leq_m B_m$

EXERCISE :

$ONE = \{ x \mid \varphi_x = \mathbb{1} \}$ not recursive

$K \leq_m ONE$

idea : given P_x construct $P_{f(x)}$ which computes $\mathbb{1}$ iff $P_x(x)\downarrow$

```
def P_f(x)(y):
    P_x(x)
    return 1
```

formally, define

$g(x,y) = \mathbb{1}(\varphi_x(x)) = \mathbb{1}(\psi_U(x,x))$ computable

then by smn theorem there is $f: \mathbb{N} \to \mathbb{N}$ total computable s.t.

$\forall x, y \qquad \varphi_{f(x)}(y) = g(x,y) = \mathbb{1}(\varphi_x(x)) = \begin{cases} 1 & \text{if } \varphi_x(x)\downarrow \\ \uparrow & \text{otherwise} \end{cases}$

Then f is the reduction function for $K \leq_m ONE$

* if $x \in K$ then $\forall y \quad \varphi_{f(x)}(y) = \mathbb{1}(\varphi_x(x)) = 1$ hence $f(x) \in ONE$

* if $x \notin K$ then $\forall y \quad \varphi_{f(x)}(y) = \mathbb{1}(\varphi_x(x)) \uparrow$ hence $\varphi_{f(x)} \neq \mathbb{1}$

  and thus $f(x) \notin ONE$

EXERCISE :

① Consider programs for URM such that jumps can only jump forward

$I_i : J(m, n, t)$ then $t > i$

can I compute all computable functions [EASY]

② Consider programs for URM such that jumps can only jump backward

$I_i : J(m, n, t)$ then $t < i$

can I compute all computable functions [FOR THE BRAVES]