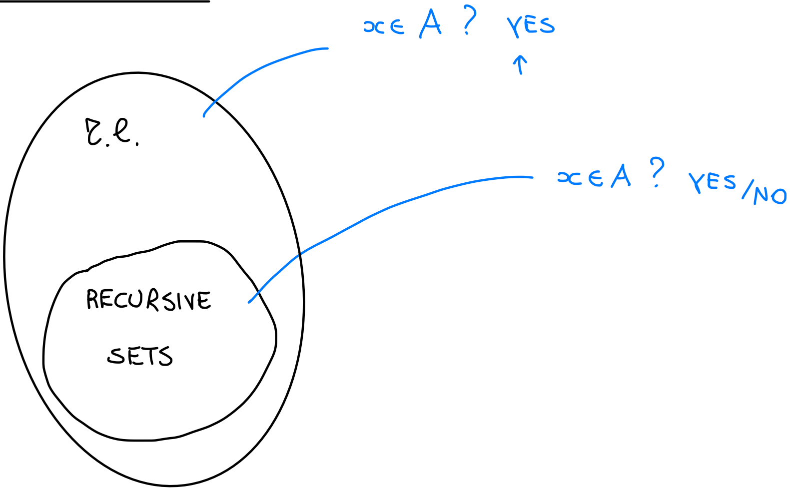
COMPUTABILITY (01/12/2025)

RECURSIVELY ENUMERABLE SETS



Def. (r.e. set) A set $A \subseteq \mathbb{N}$ is recursively enumerable (r.e.) if its semi-characteristic function $sc_A : \mathbb{N} \to \mathbb{N}$

$$sc_A(x) = \begin{cases} 1 & \text{if } x \in A \\ \uparrow & \text{otherwise} \end{cases} \quad \text{is computable}$$

A property $Q(\vec{x}) \subseteq \mathbb{N}^k$ is <u>semi-decidable</u> if

$$sc_Q(\vec{x}) = \begin{cases} 1 & \text{if } Q(\vec{x}) \\ \uparrow & \text{otherwise} \end{cases} \quad \text{is computable}$$

<u>OBSERVATION</u>: if $Q(x) \subseteq \mathbb{N}$

$Q(x)$ is semi-decidable if and only if $\{x \mid Q(x)\}$ is r.e.

(we could define what a r.e. set $A \subseteq \mathbb{N}^k$ ....)

OBSERVATION: Let A be a set ($A \subseteq \mathbb{N}$)

A recursive $\iff$ $A, \bar{A}$ r.e.

<u>proof</u>

($\Rightarrow$) Let A be recursive, i.e.

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

we want to show that A is r.e., i.e.

$$SC_A(x) = \begin{cases} 1 & \text{if } x \in A \\ \uparrow & \text{otherwise} \end{cases} \quad \text{is computable}$$

Given a program P_{χ_A}, intuitively we can define

```
def P_SCA (x) :
    if P_χA (x) = 1
        return 1
    else
        loop
```

Formally

$$SC_A(x) = \underbrace{\mathbb{1}\Big(\underbrace{\mu w.\ \underbrace{|\chi_A(x) - 1|}_{\begin{cases} 0 & \text{if } x\in A \\ 1 & \text{otherwise}\end{cases}}}_{\begin{cases} 0 & \text{if } x\in A \\ \uparrow & \text{otherwise}\end{cases}}\Big)}_{\begin{cases} 1 & \text{if } x\in A \\ \uparrow & \text{otherwise}\end{cases}}$$

computable

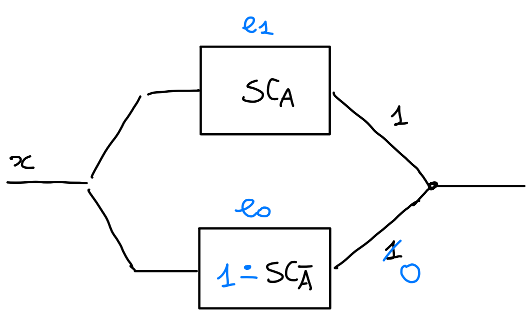
(composition & minimalisation of computable function)

hence A is r.e.

Concerning $\bar{A}$, just observe that $\bar{A}$ is recursive (since A is) hence by the above $\bar{A}$ is r.e..

($\Leftarrow$) Let $A, \bar{A}$ be r.e., i.e.

$$SC_A(x) = \begin{cases} 1 & \text{if } x \in A \\ \uparrow & \text{otherwise} \end{cases} \qquad 1 \dot{-} SC_{\bar{A}}(x) = \begin{cases} \cancel{1}\ 0 & \text{if } x \notin A \\ \uparrow & \text{otherwise} \end{cases}$$



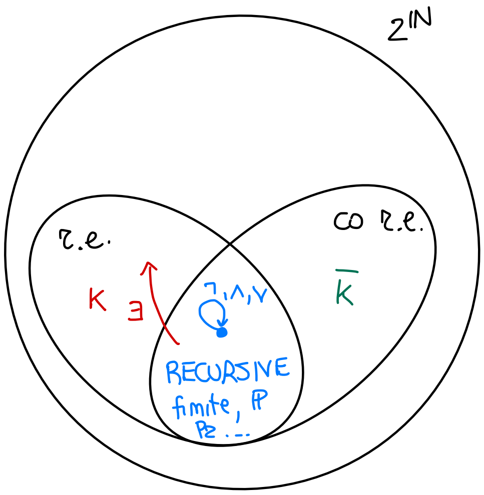
Let $e_1, e_0 \in \mathbb{N}$ be such that $\varphi_{e_1} = sc_A$ and $\varphi_{e_0} = 1 \dot{-} sc_{\bar{A}}$

$$\chi_A(x) = \left(\mu(y,t) \,.\, S(e_1, x, y, t) \vee S(e_0, x, y, t)\right)_{\downarrow y}$$

$$= \left(\mu\, \omega . \quad S(e_1, x, (\omega)_1, (\omega)_2) \vee S(e_0, x, (\omega)_1, (\omega)_2)\right)_1$$

$(\omega)_1 = y$
$(\omega)_2 = t$

computable, hence A is recursive. $\square$



co r.e. $\equiv$ sets whose complement is r.e.

* <u>K is r.e. but not recursive</u>

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

$$= \mathbb{1}(\varphi_x(x))$$

$$= \mathbb{1}(\psi_U(x, x))$$

* <u>$\bar{K}$ is not r.e.</u>

otherwise $K, \bar{K}$ r.e. $\Rightarrow$ K recursive

* <u>Existential quantification</u>

$Q(t, \vec{x}) \subseteq \mathbb{N}^{k+1}$ decidable

what about $P(\vec{x}) \equiv \exists t .\, Q(t, \vec{x})$

only semi-decidable in general

STRUCTURE THEOREM

Let $P(\vec{x}) \subseteq \mathbb{N}^k$ be a predicate. Then

$P(\vec{x})$ semi decidable $\iff$ there is $Q(t,\vec{x}) \subseteq \mathbb{N}^{k+1}$ decidable such that $P(\vec{x}) \equiv \exists t.\ Q(t,\vec{x})$

proof

$(\Rightarrow)$ Let $P(\vec{x}) \subseteq \mathbb{N}^k$ semi-decidable, i.e.

$$sc_P(\vec{x}) = \begin{cases} 1 & \text{if } P(\vec{x}) \\ \uparrow & \text{otherwise} \end{cases} \quad \text{computable}$$

i.e. there is $e \in \mathbb{N}$ s.t. $sc_P = \varphi_e^{(k)}$

Now observe that $\forall \vec{x} \in \mathbb{N}^k$

$$P(\vec{x}) \quad \text{iff} \quad \underset{\varphi_e^{(k)}}{\overset{\parallel}{sc_P}}(\vec{x}) = 1 \quad \text{iff} \quad \exists t.\ S^{(k)}(e,\vec{x},1,t)$$

If we define $Q(t,\vec{x}) \equiv S^{(k)}(e,\vec{x},1,t)$

then $Q(t,x)$ is decidable and

$$P(\vec{x}) \equiv \exists t.\ Q(t,\vec{x})$$

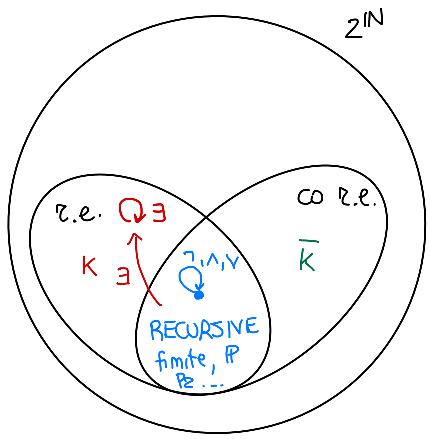
$(\Leftarrow)$ Let $P(\vec{x}) \equiv \exists t.\ Q(t,\vec{x})$ with $Q(t,\vec{x})$ decidable

Then

$$sc_P(\vec{x}) = \begin{cases} 1 & \text{if } P(\vec{x}) \leftrightarrow \exists t.\ Q(t,\vec{x}) \leftrightarrow \exists t.\ \chi_Q(t,\vec{x}) = 1 \\ \uparrow & \text{otherwise} \end{cases}$$

$$= \mathbb{1}(\mu t.\ |\chi_Q(t,\vec{x}) - 1|)$$

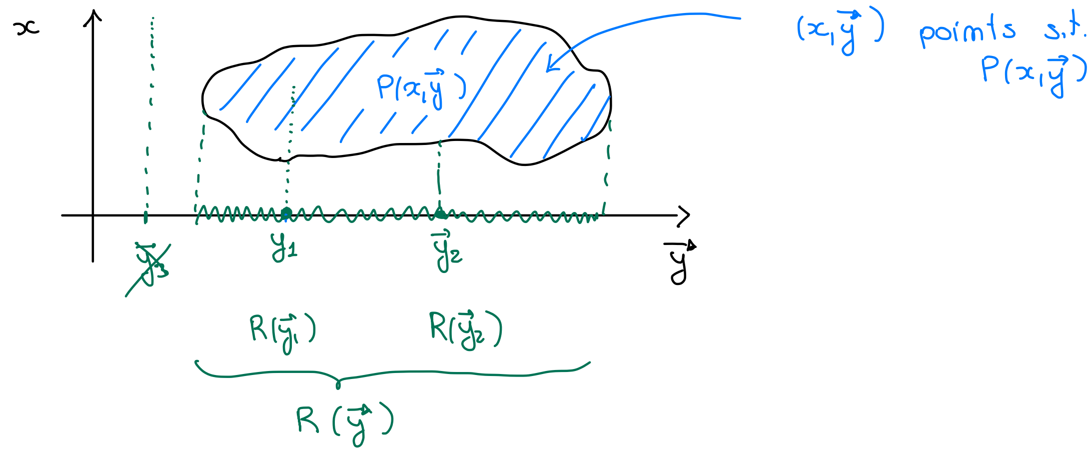
computable, hence $P(\vec{x})$ is semi-decidable $\square$



Projection Theorem

Let $P(x,\vec{y}) \subseteq \mathbb{N}^{k+1}$ semi-decidable.

Then also $R(\vec{y}) \equiv \exists x.\ P(x,\vec{y})$ is semi-decidable



proof

Let $P(x,\vec{y})$ semi-decidable. By the structure theorem

$$P(x,\vec{y}) \equiv \exists t.\ Q(t,x,\vec{y}) \qquad \text{with } Q(t,x,\vec{y}) \subseteq \mathbb{N}^{k+2} \text{ decidable}$$

Then

$$R(\vec{y}) \equiv \exists x.\ P(x,\vec{y}) \equiv \exists x.\ \exists t.\ Q(t,x,\vec{y})$$

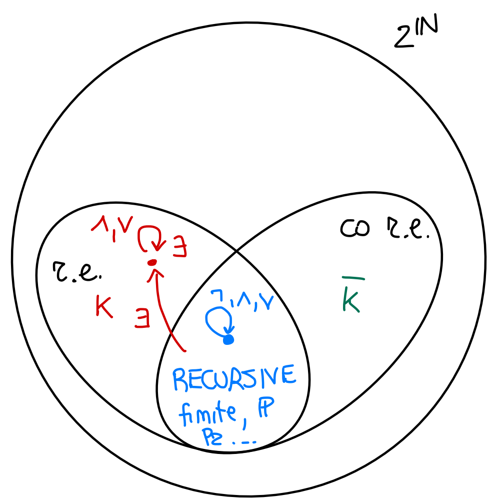
$$\equiv \exists \omega.\ Q((\omega)_1, (\omega)_2, \vec{y})$$

if we let $Q'(\omega,\vec{y}) \equiv Q((\omega)_1,(\omega)_2,\vec{y})$ decidable

we have

$$R(\vec{y}) \equiv \exists \omega.\ Q'(\omega,\vec{y})$$

Thus, again by the structure theorem, $R(\vec{y})$ is semi-decidable $\square$



OBSERVATION : Let $P(\vec{x}), Q(\vec{x}) \subseteq \mathbb{N}^k$ semi-decidable. Then

(1) $P(\vec{x}) \wedge Q(\vec{x})$

(2) $P(\vec{x}) \vee Q(\vec{x})$

are semi-decidable

proof

By the structure theorem

$$P(\vec{x}) \equiv \exists t.\ P'(t, \vec{x}) \qquad P'(t,\vec{x}) \text{ decidable}$$

$$Q(\vec{x}) \equiv \exists t.\ Q'(t, \vec{x}) \qquad Q'(t,\vec{x}) \quad \text{"}$$

Then

(1)
$$P(\vec{x}) \wedge Q(\vec{x}) \equiv \exists t_1.\ P'(t_1, \vec{x}) \wedge \exists t_2.\ Q'(t_2, \vec{x})$$

$$\equiv \exists \omega.\ \big(\underbrace{P'((\omega)_1, \vec{x}) \wedge Q'((\omega)_2, \vec{x})}_{\text{decidable}}\big)$$

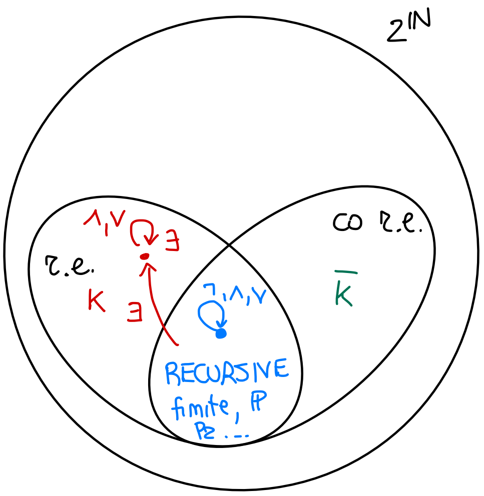
hence, by the structure theorem, $P(\vec{x}) \wedge Q(\vec{x})$ is semi-decidable.

(2)
$$P(\vec{x}) \vee Q(\vec{x}) \equiv \exists t.\ P'(t, \vec{x}) \vee \exists t.\ Q'(t, \vec{x})$$

$$\equiv \exists t.\ \big(\overbrace{P'(t, \vec{x}) \vee Q'(t, \vec{x})}^{\text{decidable}}\big)$$

hence, by the structure theorem, $P(\vec{x}) \vee Q(\vec{x})$ is semi-decidable.

$\square$



* NEGATION

$Q(x) \equiv$ " $x \in K$ " semi-decidable

but

$\neg Q(x) \equiv$ " $x \notin K$ " $\equiv$ " $x \in \overline{K}$ "

not semidecidable

* UNIVERSAL QUANTIFICATION

consider $R(t,x) \equiv \neg H(x,x,t)$ decidable

"$x \in \bar{K}$" $\equiv \forall t.\ R(t,x) \equiv \forall t.\ \neg H(x,x,t)$ non r.e. /

non semi-decidable

EXERCISE

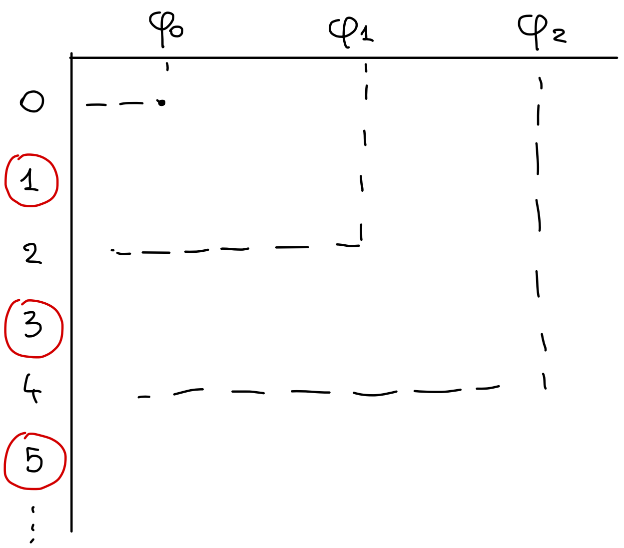
Is it the case that if $\exists t.\ Q(x,t)$ is semi-decidable

then $Q(x,t)$ is semi-decidable ? [TRY]

EXERCISE: Define a function $f: \mathbb{N} \to \mathbb{N}$ total and non-computable

s.t. $f(x) = x$ on infinitely many inputs (i.e. $\{x \mid f(x) = x\}$ infinite)

1st idea :



$$f(x) = \begin{cases} x & \text{if } x \text{ is odd} \\ \varphi_{\frac{x}{2}}(x) + 1 & \text{if } \varphi_{x/2}(x) \downarrow \\ 0 & \text{if } \varphi_{x/2}(x) \uparrow \end{cases}$$

- f is total
- $\{x \mid f(x) = x\} \supseteq$ odd numbers , hence it is infinite
- f is not computable since it is total and different from all total computable functions.

  $\forall x$ if φ_x is total $\Rightarrow \varphi_x(2x)\downarrow$ hence

  $f(2x) = \varphi_x(2x) + 1 \neq \varphi_x(2x)$

2nd idea

$$f(x) = \begin{cases} \varphi_x(x) + 1 & \text{if } \varphi_x(x)\downarrow \\ x & \text{if } \varphi_x(x)\uparrow \end{cases}$$

- f is total
- f is not computable (different from all total computable functions)

  ($\forall x$ if φ_x is total $f(x) = \varphi_x(x) + 1 \neq \varphi_x(x)$)

- $\{x \mid f(x) = x\} \supseteq \{x \mid \varphi_x(x)\uparrow\} = \bar{K}$ infinite

  (e.g. because if it were finite, it would be recursive, while it is not)

3rd idea

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

- f is total
- $\{x \mid f(x) = x\} \supseteq \bar{K}$ infinite
- f is not computable

  $\chi_K(x) = f(x) \dot{-} x$

hence, if f were computable, also χ_K would be.

EXERCISE : If f is computable

and $g(x) = f(x)$ almost everywhere ($\{x \mid g(x) \neq f(x)\}$ is finite)

$\Rightarrow$ g computable.