WS 2023 lecture 3

Computability Theory

Aart Middeldorp

Outline

Definitions

- $\langle x_1, \ldots, x_n \rangle = \mathrm{p}_0^n \times \mathrm{p}_1^{x_1} \times \cdots \times \mathrm{p}_n^{x_n}$ Gödel number
- $(x)_i = (\mu\, j < x)\ \neg(p_i^{j+1} \mid x)$
- $\mathrm{len}(x) = (x)_0$
- $\mathrm{seq}(x) \iff x > 0 \wedge (\forall\, i \leqslant x)\ [\ (x)_i \neq 0 \implies i \leqslant \mathrm{len}(x)\ ]$
- $x\ ;\ y = \mathrm{p}_0^{\mathrm{len}(x)+\mathrm{len}(y)} \times \prod_{i < \mathrm{len}(x)} \mathrm{p}_{i+1}^{(x)_{i+1}} \times \prod_{i < \mathrm{len}(y)} \mathrm{p}_{\mathrm{len}(x)+i+1}^{(y)_{i+1}}$

Lemma

PR is closed under course–of–values recursion:
if $g\colon \mathbb{N}^n \to \mathbb{N}$ and $h\colon \mathbb{N}^{n+2} \to \mathbb{N}$ are primitive recursive then

$$f(0, \vec{y}) = g(\vec{y}) \qquad\qquad f(x+1, \vec{y}) = h(\tilde{f}(x, \vec{y}), x, \vec{y})$$

with $\tilde{f}(x, \vec{y}) = \langle f(0, \vec{y}), \ldots, f(x, \vec{y}) \rangle$ is primitive recursive

Definition (Ackermann function)

$$\begin{aligned} \mathrm{ack}(0, y) &= y + 1 \\ \mathrm{ack}(x+1, 0) &= \mathrm{ack}(x, 1) \\ \mathrm{ack}(x+1, y+1) &= \mathrm{ack}(x, \mathrm{ack}(x+1, y)) \end{aligned}$$

Lemma

$\forall$ primitive recursive function $f\colon \mathbb{N}^n \to \mathbb{N}$ $\exists$ constant $c \in \mathbb{N}$ such that

$$f(x_1, \ldots, x_n) < \mathrm{ack}(c, \max\{x_1, \ldots, x_n\})$$

Theorem

Ackermann function is not primitive recursive

Definition

class R of recursive functions is smallest class of total functions that contains all initial functions and is closed under composition, primitive recursion and (unbounded) minimization:

$$(\mu\, i)\, P(i, \vec{y}) = \min\{i \mid P(i, \vec{y})\} \in \mathrm{R}$$

for all $P\colon \mathbb{N}^{n+1} \to \mathbb{B}$ such that $\chi_P \in \mathrm{R}$ and $\forall\, \vec{y}\; \exists\, x\; P(x, \vec{y})$

Theorem

- $\exists$ primitive recursive predicate $P\colon \mathbb{N} \to \mathbb{B}$ such that

$$P(x) \iff x \text{ encodes correct Ackermann computation tree}$$

- $\mathrm{ack}(x, y) = z \iff \exists\, t$ such that $P(t)$ and $(t)_1 = \langle x, y, z\rangle$
- $\mathrm{ack}(x, y) = ((\mu\, t)\, (P(t) \wedge (t)_{1,1} = x \wedge (t)_{1,2} = y))_{1,3}$
- Ackermann function is recursive

Definition

index $\ulcorner f \urcorner \in \mathbb{N}$ of derivation of primitive recursive function f is defined inductively:

- $\ulcorner \mathrm{z} \urcorner = \langle 0 \rangle$
- $\ulcorner \mathrm{s} \urcorner = \langle 1 \rangle$
- $\ulcorner \pi^n_i \urcorner = \langle 2, n, i \rangle$
- $\ulcorner f \urcorner = \langle 3, \ulcorner g \urcorner, \ulcorner h_1 \urcorner, \dots, \ulcorner h_m \urcorner \rangle$ if f is obtained by composing g and $h_1, \dots, h_m$
- $\ulcorner f \urcorner = \langle 4, \ulcorner g \urcorner, \ulcorner h \urcorner \rangle$ if f is obtained by primitive recursion from g and h

Part I: Recursive Function Theory

Ackermann function, bounded minimization, bounded recursion, course–of–values recursion, diagonalization, diophantine sets, elementary functions, fixed point theorem, Fibonacci numbers, Gödel numbering, Gödel's β function, Grzegorczyk hierarchy, loop programs, minimization, normal form theorem, partial recursive functions, primitive recursion, recursive enumerability, recursive inseparability, s–m–n theorem, total recursive functions, undecidability, while programs, ...

Part II: Combinatory Logic and Lambda Calculus

α–equivalence, abstraction, arithmetization, β–reduction, CL–representability, combinators, combinatorial completeness, Church numerals, Church–Rosser theorem, Curry–Howard isomorphism, de Bruijn notation, η–reduction, fixed point theorem, intuitionistic propositional logic, λ–definability, normalization theorem, termination, typing, undecidability, Z property, ...

Outline

Loop Programs

simple programming language

- natural numbers are only data type
- variables x, y, z, ...
- commands
  - assignment $x := 0$ $x := y$
  - increment $x{+}{+}$
  - composition $P\,;\,Q$
  - loop
    - LOOP x DO P OD

  execute P exactly n times, where n is value of x before entering loop

Examples

program P:

```
z := x ;
LOOP y DO
    z++
OD
```

computes addition: $z = x + y$

notation: $P(x, y; z)$

multiplication $(x, y; z)$:

```
z := 0 ;
LOOP x DO
    LOOP y DO
        z++
    OD
OD
```


Examples

program $P(x; y)$:

```
y := 0 ;
z := 0 ;
LOOP x DO
    y := z ;
    z++
OD
```

computes predecessor: $y = \mathrm{p}(x)$

program $Q(x; y)$:

```
y := 0 ;
z := 0 ;
z++ ;
LOOP x DO
    y := z
OD
```

computes sign function: $y = \mathrm{sg}(x)$

Lemma

LOOP programs terminate

Definition

function $f \colon \mathbb{N}^n \to \mathbb{N}$ is LOOP computable if $\exists$ LOOP program $P(x_1, \ldots, x_n; y)$ such that

$$y = f(x_1, \ldots, x_n)$$

after execution of P

Theorem

primitive recursive functions are LOOP computable

Proof

- zero $\quad y = \mathsf{z}(x) \quad$ $y := 0$
- successor $\quad y = \mathsf{s}(x) \quad$ $y := x\,;\ y{+}{+}$
- projection $\quad y = \pi_i^n(x_1,\ldots,x_n) \quad$ $y := x_i$
- composition $\quad y = g(h_1(x_1,\ldots,x_m),\ldots,h_n(x_1,\ldots,x_m))$

  $P_{h_1}(x_1,\ldots,x_m; y_1)\,;\ \ldots\,;\ P_{h_n}(x_1,\ldots,x_m; y_n)\,;\ P_g(y_1,\ldots,y_n; y)$

- primitive recursion

$$f(0,\vec{y}) = g(\vec{y})$$
$$f(x+1,\vec{y}) = h(f(x,\vec{y}),x,\vec{y})$$

```
Pg(y1, ..., yn; z);
v := 0;
LOOP x DO
    Ph(z, v, y1, ..., yn; w);
    z := w;
    v++
OD
```


Theorem

LOOP computable functions are primitive recursive

Proof

- LOOP program P with variables $x_1,\ldots,x_n$
- claim: $\exists$ primitive recursive functions $f_1,\ldots,f_n\colon \mathbb{N}^n \to \mathbb{N}$ such that

  if $x_1,\ldots,x_n$ have values $a_1,\ldots,a_n$ before execution of P

  then x_i has value $f_i(a_1,\ldots,a_n)$ after execution of P, for all $1 \leqslant i \leqslant n$
- claim is proved by induction on structure of P (details left as exercise)

Example

$f_P(x,y) = f_3(x,y,0,0,0)$ for LOOP program P with input variables x_1 and x_2, output variable x_3, and auxiliary variables x_4 and x_5

Outline

Definition

class E of elementary functions is smallest class of (total) functions $f\colon \mathbb{N}^n \to \mathbb{N}$ that contains all initial functions, $+$, $\dot{-}$ and is closed under composition, bounded summation and bounded product

Examples

$x \times y$, x^y and $x!$ are elementary functions:

- $x \times y = \sum_{i<x} y = \sum_{i<x} \pi_2^2(i,y)$
- $x^y = \prod_{i<y} x = \prod_{i<y} \pi_2^2(i,x)$
- $x! = \prod_{i<x} i+1 = \prod_{i<x} \mathsf{s}(i)$

Lemma

E is closed under bounded minimization

Proof

- consider $f(x,\vec{y}) = (\mu\, i \leqslant x)\ (g(i,\vec{y}) = 0)$ with $g \in \mathrm{E}$
- elementary function

$$f'(x,\vec{y}) = \sum_{i \leqslant x} 1 \dot{-} g(i,\vec{y})$$

  counts how many values $i \in \{0,\dots,x\}$ satisfy $g(i,\vec{y}) = 0$
- $f(x,\vec{y}) = \sum_{i \leqslant x} 1 \dot{-} f'(i,\vec{y})$ is elementary

Lemma

elementary functions are primitive recursive

Definition

binary function $2_x(y)$ is defined by primitive recursion

$$2_0(y) = y \qquad\qquad 2_{x+1}(y) = 2^{2_x(y)}$$

Example

$2_0(3) = 3 \qquad 2_1(3) = 2^3 = 8 \qquad 2_2(3) = 2^8 = 256$

Lemma

$\forall$ elementary function $f\colon \mathbb{N}^n \to \mathbb{N}$ $\exists$ constant $c \in \mathbb{N}$ such that

$$f(x_1,\dots,x_n) < 2_c(\max\{x_1,\dots,x_n\})$$

Proof

induction on definition of elementary functions

- initial functions

  $z(x) = 0 < 2^x = 2_1(x)$

  $s(x) = x + 1 < 2^{2^x} = 2_2(x)$

  $\pi_i^n(x_1,\dots,x_n) = x_i \leqslant \max\{x_1,\dots,x_n\} < 2_1(\max\{x_1,\dots,x_n\})$
- $+$ and $\dot{-}$ $\qquad x \dot{-} y \leqslant x + y \leqslant 2 \times \max\{x,y\} < 2_2(\max\{x,y\})$
- composition $\quad f(\vec{x}) = g(h_1(\vec{x}),\dots,h_m(\vec{x}))$

  $g(\vec{y}) < 2_c(\max\{\vec{y}\}) \qquad h_i(\vec{x}) < 2_{c_i}(\max\{\vec{x}\})$ for all $1 \leqslant i \leqslant m$

$$\begin{aligned} f(\vec{x}) &< 2_c(\max\{h_1(\vec{x}),\dots,h_m(\vec{x})\}) < 2_c(\max\{2_{c_1}(\max\{\vec{x}\}),\dots,2_{c_m}(\max\{\vec{x}\})\}) \\ &= 2_c(2_{\max\{c_1,\dots,c_m\}}(\max\{\vec{x}\})) = 2_{c+\max\{c_1,\dots,c_m\}}(\max\{\vec{x}\}) \end{aligned}$$

Proof (cont'd)

induction on definition of elementary functions

- bounded summation $\quad f(x,\vec{y}) = \sum_{i=0}^{x} g(i,\vec{y})$ with $g(i,\vec{y}) < 2_c(\max\{i,\vec{y}\})$

$$f(x,\vec{y}) < (x+1) \times 2_c(\max\{x,\vec{y}\}) \leqslant 2_2(\max\{x,\vec{y}\}) \times 2_c(\max\{x,\vec{y}\}) \leqslant 2_{c+3}(\max\{x,\vec{y}\})$$

Lemma

$x^y \leqslant 2_3(\max\{x,y\})$

Proof

$$x^y \leqslant 2^{xy} \leqslant 2^{2^{x+y}} \leqslant 2^{2^{2^{\max\{x,y\}}}} = 2_3(\max\{x,y\})$$

Proof (cont'd)

induction on definition of elementary functions

- bounded product $f(x,\vec{y}) = \prod_{i=0}^{x} g(i,\vec{y})$ with $g(i,\vec{y}) < 2_c(\max\{i,\vec{y}\})$

$$\begin{aligned} f(x,\vec{y}) &< 2_c(\max\{x,\vec{y}\})^{x+1} \leqslant 2_c(\max\{x,\vec{y}\})^{2_1(\max\{x,\vec{y}\})} \\ &\leqslant 2_3(\max\{2_c(\max\{x,\vec{y}\}), 2_1(\max\{x,\vec{y}\})\}) = 2_3(2_{\max\{c,1\}}(\max\{x,\vec{y}\})) \\ &= 2_{3+\max\{c,1\}}(\max\{x,\vec{y}\}) \end{aligned}$$

Corollary

$\mathrm{E} \subsetneq \mathrm{PR}$

Proof

if primitive recursive function $2_x(y)$ is elementary then $2_x(y) < 2_c(\max\{x,y\})$ for some constant c and thus $2_c(c) < 2_c(\max\{c,c\}) = 2_c(c)$ ↯

Outline

Definition (bounded recursion)

class C of numeric functions is closed under bounded recursion if $f\colon \mathbb{N}^{n+1} \to \mathbb{N}$ defined by primitive recursion from $g\colon \mathbb{N}^n \to \mathbb{N} \in \mathrm{C}$ and $h\colon \mathbb{N}^{n+2} \to \mathbb{N} \in \mathrm{C}$ and satisfying

$$f(x,\vec{y}) \leqslant i(x,\vec{y})$$

for some $i\colon \mathbb{N}^{n+1} \to \mathbb{N} \in \mathrm{C}$ different from f belongs to C

Definitions

- $\mathrm{e}_0(x,y) = x + y \qquad \mathrm{e}_1(x) = x^2 + 2$
- $\mathrm{e}_{n+2}(x) = \begin{cases} 2 & \text{if } x = 0 \\ \mathrm{e}_{n+1}(\mathrm{e}_{n+2}(x-1)) & \text{if } x > 0 \end{cases}$

Example

$\mathrm{e}_2(0) = 2 \qquad \mathrm{e}_2(1) = 6 \qquad \mathrm{e}_2(3) = \mathrm{e}_1(\mathrm{e}_2(2)) = \mathrm{e}_1(38) = 1446$

Definition

$$\mathrm{e}_{n+2}(x) = \begin{cases} 2 & \text{if } x = 0 \\ \mathrm{e}_{n+1}(\mathrm{e}_{n+2}(x-1)) & \text{if } x > 0 \end{cases} \qquad f^{(x)}(y) = \begin{cases} y & \text{if } x = 0 \\ f(f^{(x-1)}(y)) & \text{if } x > 0 \end{cases}$$

Lemma

$\mathrm{e}_{n+2}(x) = \mathrm{e}_{n+1}^{(x)}(2)$

Proof

induction on x

- $\mathrm{e}_{n+2}(0) = 2 = \mathrm{e}_{n+1}^{(0)}(2)$
- $\mathrm{e}_{n+2}(x+1) = \mathrm{e}_{n+1}(\mathrm{e}_{n+2}(x)) = \mathrm{e}_{n+1}(\mathrm{e}_{n+1}^{(x)}(2)) = \mathrm{e}_{n+1}^{(x+1)}(2)$

Definitions

- E_0 is smallest class of functions that contains all initial functions and is closed under composition and bounded recursion
- E_{n+1} is smallest class of functions that contains all initial functions, e_0, e_n and is closed under composition and bounded recursion

Examples

- E_0 contains $f(x) = x + 2$
- E_1 contains $f(x) = 4x$
- E_2 contains $f(x) = x^4$
- E_3 contains $f(x) = 2^{2^x}$

Lemma

$\forall\, f\colon \mathbb{N}^n \to \mathbb{N} \in \mathrm{E}_0 \quad \exists\, c \in \mathbb{N} \quad$ such that $\quad f(x_1,\dots,x_n) \leqslant \max\{x_1,\dots,x_n\} + c$

Proof

- $\mathrm{z}(x) \leqslant x$
- $\mathrm{s}(x) \leqslant x + 1$
- $\pi_i^n(x_1,\dots,x_n) \leqslant x_i \leqslant \max\{x_1,\dots,x_n\}$
- f is obtained by composing g and $h_1,\dots,h_m$

$$\begin{aligned} f(\vec{x}) = g(h_1(\vec{x}),\dots,h_m(\vec{x})) &\leqslant \max\{h_1(\vec{x}),\dots,h_m(\vec{x})\} + c_g && g \in \mathrm{E}_0 \\ &\leqslant \max\{\vec{x}\} + \max\{c_{h_1},\dots,c_{h_m}\} + c_g && h_1,\dots,h_m \in \mathrm{E}_0 \end{aligned}$$

- f is obtained by bounded recursion from g and h with bound i

$$f(\vec{x}) \leqslant i(\vec{x}) \leqslant \max\{x_1,\dots,x_n\} + c_i \qquad i \in \mathrm{E}_0$$

Lemma

$\forall\, f\colon \mathbb{N}^n \to \mathbb{N} \in \mathrm{E}_1 \quad \exists$ linear function $g\colon \mathbb{N}^n \to \mathbb{N} \quad$ such that $\quad f(x_1,\dots,x_n) \leqslant g(x_1,\dots,x_n)$

Proof

- $\cdots$
- $f = \mathrm{e}_0 = x + y$
- f is obtained by composing g and $h_1,\dots,h_m$

$$\begin{aligned} f(\vec{x}) = g(h_1(\vec{x}),\dots,h_m(\vec{x})) &\leqslant \ell_g(h_1(\vec{x}),\dots,h_m(\vec{x})) && g \in \mathrm{E}_1 \\ &\leqslant \ell_g(\ell_{h_1}(\vec{x}),\dots,\ell_{h_m}(\vec{x})) && h_1,\dots,h_m \in \mathrm{E}_1 \end{aligned}$$

linear functions are closed under composition

Lemma

1. $\mathrm{e}_n(x) > x \qquad \forall\, n \geqslant 1\ \forall\, x \geqslant 0$
2. $\mathrm{e}_n(x+1) > \mathrm{e}_n(x) \qquad \forall\, n \geqslant 1\ \forall\, x \geqslant 0$
3. $\mathrm{e}_{n+1}(x) \geqslant \mathrm{e}_n(x) \qquad \forall\, n \geqslant 1\ \forall\, x \geqslant 0$
4. $\mathrm{e}_{n+1}(x+k) \geqslant \mathrm{e}_n^{(k)}(x) \qquad \forall\, n \geqslant 1\ \forall\, x \geqslant 0\ \forall\, k \geqslant 0$

Proof 1

induction on $n > 0$

- $\mathrm{e}_1(x) = x^2 + 2 > x$
- induction on $x \geqslant 0$
  - $\mathrm{e}_{n+1}(0) = 2 > 0$
  - $\mathrm{e}_{n+1}(x+1) = \mathrm{e}_n(\mathrm{e}_{n+1}(x)) > \mathrm{e}_{n+1}(x) > x \quad \Longrightarrow \quad \mathrm{e}_{n+1}(x+1) > x + 1$

Lemma

1. $\mathrm{e}_n(x) > x$ $\quad \forall\, n \geqslant 1\ \forall\, x \geqslant 0$
2. $\mathrm{e}_n(x+1) > \mathrm{e}_n(x)$ $\quad \forall\, n \geqslant 1\ \forall\, x \geqslant 0$
3. $\mathrm{e}_{n+1}(x) \geqslant \mathrm{e}_n(x)$ $\quad \forall\, n \geqslant 1\ \forall\, x \geqslant 0$
4. $\mathrm{e}_{n+1}(x+k) \geqslant \mathrm{e}_n^{(k)}(x)$ $\quad \forall\, n \geqslant 1\ \forall\, x \geqslant 0\ \forall\, k \geqslant 0$

Proof 2

induction on $n > 0$

- $\mathrm{e}_1(x+1) = (x+1)^2 + 2 > x^2 + 2 = \mathrm{e}_1(x)$
- $\mathrm{e}_{n+1}(x+1) = \mathrm{e}_n(\mathrm{e}_{n+1}(x)) > \mathrm{e}_{n+1}(x)$ by 1

Proof 3 4

homework exercise

Lemma

$\forall\, f\colon \mathbb{N}^m \to \mathbb{N} \in \mathrm{E}_{n+2} \quad \exists\, k \in \mathbb{N}$ such that $f(x_1,\dots,x_m) \leqslant \mathrm{e}_{n+1}^{(k)}(\max\{x_1,\dots,x_m\})$

Proof

- $\mathrm{z}(x) = 0 \leqslant x = \mathrm{e}_{n+1}^{(0)}(x)$
- $\mathrm{s}(x) = x+1 \leqslant x^2+2 = \mathrm{e}_1(x) \leqslant \mathrm{e}_{n+1}^{(1)}(x)$
- $\pi_i^n(x_1,\dots,x_n) = x_i < \mathrm{e}_1(x_i) \leqslant \mathrm{e}_1(\max\{\vec{x}\}) \leqslant \mathrm{e}_{n+1}^{(1)}(\max\{\vec{x}\})$
- $\mathrm{e}_0(x,y) = x+y \leqslant 2 \times \max\{x,y\} \leqslant (\max\{x,y\})^2 + 2 = \mathrm{e}_1(\max\{x,y\}) \leqslant \mathrm{e}_{n+1}^{(1)}(\max\{x,y\})$
- $\mathrm{e}_{n+1}(x) \leqslant \mathrm{e}_{n+1}^{(1)}(x)$

Lemma

$\forall\, f\colon \mathbb{N}^m \to \mathbb{N} \in \mathrm{E}_{n+2} \quad \exists\, k \in \mathbb{N}$ such that $f(x_1,\dots,x_m) \leqslant \mathrm{e}_{n+1}^{(k)}(\max\{x_1,\dots,x_m\})$

Proof (cont'd)

- f is obtained by composing g and $h_1,\dots,h_j$

$$\begin{aligned} f(\vec{x}) &= g(h_1(\vec{x}),\dots,h_j(\vec{x})) \leqslant \mathrm{e}_{n+1}^{(k)}(\max\{h_1(\vec{x}),\dots,h_j(\vec{x})\}) && g \in \mathrm{E}_{n+2} \\ &\leqslant \mathrm{e}_{n+1}^{(k)}(\mathrm{e}_{n+1}^{(\ell)}(\max\{\vec{x}\})) = \mathrm{e}_{n+1}^{(k+\ell)}(\max\{\vec{x}\}) \end{aligned}$$

$$h_i(\vec{x}) \leqslant \mathrm{e}_{n+1}^{(k_i)}(\max\{\vec{x}\}) \leqslant \mathrm{e}_{n+1}^{(\ell)}(\max\{\vec{x}\}) \qquad h_1,\dots,h_j \in \mathrm{E}_{n+2}$$

$\ell = \max\{k_1,\dots,k_j\}$

- f is obtained by bounded recursion from g and h with bound i

$$f(\vec{x}) \leqslant i(\vec{x}) \leqslant \mathrm{e}_{n+1}^{(k)}(\max\{\vec{x}\}) \qquad i \in \mathrm{E}_{n+2}$$

Lemma

$\forall\, f\colon \mathbb{N}^m \to \mathbb{N} \in \mathrm{E}_{n+2} \quad \exists\, k \in \mathbb{N}$ such that $f(x_1,\dots,x_m) \leqslant \mathrm{e}_{n+1}^{(k)}(\max\{x_1,\dots,x_m\})$

Corollary

$\forall\, f\colon \mathbb{N} \to \mathbb{N} \in \mathrm{E}_{n+2} \quad \exists\, k \in \mathbb{N}$ such that $f(x) \leqslant \mathrm{e}_{n+2}(k+x)$

Proof

$f(x) \leqslant \mathrm{e}_{n+1}^{(k)}(x) \leqslant \mathrm{e}_{n+2}(k+x)$

Lemma

$e_n \in E_{n+1} \setminus E_n$

Proof

$e_n \notin E_n$ by case analysis on n

- $e_0(x,y) = x + y \implies \neg\exists c \in \mathbb{N}$ with $x + y \leqslant \max\{x,y\} + c$
- $e_1(x) = x^2 + 2 \implies \neg\exists a,b \in \mathbb{N}$ with $x^2 + 2 \leqslant ax + b$
- $e_{n+2} \in E_{n+2} \implies e_{n+2}(e_0(x,x)) \in E_{n+2}$ and thus

$$e_{n+2}(x + x) \leqslant e_{n+2}(k + x)$$

for some $k \in \mathbb{N}$ ↯

Theorem (Grzegorczyk Hierarchy)

1. $E_0 \subsetneq E_1 \subsetneq E_2 \subsetneq \cdots$
2. $E_3 = E$
3. $\bigcup_{n \geqslant 0} E_n = PR$

Proof

1. $E_0 \subseteq E_1 \subseteq E_2$ by definition
   $E_{n+2} \subseteq E_{n+3}$ for $n \geqslant 0$ because $e_{n+1} \in E_{n+3}$ by bounded recursion:

$$e_{n+1}(x) \leqslant e_{n+2}(x) \in E_{n+3}$$

Outline

Important Concepts

- $2_x(y)$
- bounded recursion
- e_0, e_1, e_2, ...
- E_0, E_1, E_2, ...
- elementary function
- Grzegorczyk hierarchy
- LOOP computable
- LOOP program

homework for October 23