Les fonctions récursives

- Un modèle de calcul : fonctions des entiers dans les entiers
- Équivalent aux machines de Turing
- Liens avec les théories de l'arithmétique

Fonctions récursives primitives

Les fonctions initiales

- les projections $P_k^i : \mathbb{N}^k \mapsto \mathbb{N}$. $P_k^i(n_1, \ldots, n_k) = n_i$
- le successeur $S(n) = n + 1$
- la fonction nulle $Z(n) = 0$
- la fonction constante 0 (0 arguments).

Composition

$\phi = \mathrm{Comp}_m(\xi, \psi_1, \ldots, \psi_m)$:

$$\phi(\vec{n}) = \xi(\psi_1(\vec{n}), \ldots, \psi_m(\vec{n}))$$

Fonctions récursives primitives (2)

Récursion primitive

$$\phi(m, \vec{n}) = \begin{cases} \xi(\vec{n}) & \text{si } m = 0 \\ \psi(\phi(m-1, \vec{n}), m-1, \vec{n}) & \text{si } m > 0 \end{cases}$$

$\phi = \mathrm{Prim}(\xi, \psi)$

Fonctions récursives primitives (2)

Récursion primitive

$$\phi(m, \vec{n}) = \begin{cases} \xi(\vec{n}) & \text{si } m = 0 \\ \psi(\phi(m-1, \vec{n}), m-1, \vec{n}) & \text{si } m > 0 \end{cases}$$

$\phi = \mathrm{Prim}(\xi, \psi)$

Définition

L'ensemble des *fonctions récursives primitives* est le plus petit ensemble contenant les fonctions initiales, clos par composition et récursion primitive.

Fonctions récursives primitives (2)

Récursion primitive

$$\phi(m, \vec{n}) = \begin{cases} \xi(\vec{n}) & \text{si } m = 0 \\ \psi(\phi(m-1, \vec{n}), m-1, \vec{n}) & \text{si } m > 0 \end{cases}$$

$\phi = \mathrm{Prim}(\xi, \psi)$

Définition

L'ensemble des *fonctions récursives primitives* est le plus petit ensemble contenant les fonctions initiales, clos par composition et récursion primitive.

Boucles for

Les fonctions récursives primitives sont celle qu'on peut définir avec des boucles for

Exemples de fonctions récursives primitives

$$m + n = \begin{cases} n & \text{Si } m = 0 \\ S(m - 1 + n) & \text{Si } m > 0 \end{cases}$$

Exemples de fonctions récursives primitives

$$m + n = \begin{cases} n & \text{Si } m = 0 \\ S(m - 1 + n) & \text{Si } m > 0 \end{cases}$$

$$+(m, n) = \begin{cases} \xi(n) & \text{si } m = 0 \\ \psi(+(m-1, n), m-1, n) & \text{si } m > 0 \end{cases}$$

$+ = \text{Prim}(\xi, \psi)$

Exemples de fonctions récursives primitives

$$m + n = \begin{cases} n & \text{Si } m = 0 \\ S(m - 1 + n) & \text{Si } m > 0 \end{cases}$$

$$+(m, n) = \begin{cases} \xi(n) & \text{si } m = 0 \\ \psi(+(m-1, n), m-1, n) & \text{si } m > 0 \end{cases}$$

$+ = \text{Prim}(\xi, \psi)$

$\xi = P_1^1$

Exemples de fonctions récursives primitives

$$m + n = \begin{cases} n & \text{Si } m = 0 \\ S(m - 1 + n) & \text{Si } m > 0 \end{cases}$$

$$+(m, n) = \begin{cases} \xi(n) & \text{si } m = 0 \\ \psi(+(m - 1, n), m - 1, n) & \text{si } m > 0 \end{cases}$$

$+ = \text{Prim}(\xi, \psi)$

$\xi = P_1^1$
$\psi = \text{Comp}_1(S, P_3^1)$

Exemples de fonctions récursives primitives (2)

$$m \times n = \begin{cases} 0 & \text{Si } m = 0 \\ (m-1) \times n + n & \text{Si } m > 0 \end{cases}$$

Exemples de fonctions récursives primitives (2)

$$m \times n = \begin{cases} 0 & \text{Si } m = 0 \\ (m-1) \times n + n & \text{Si } m > 0 \end{cases}$$

$$\times = \mathrm{Prim}(Z, \mathrm{Comp}_2(+, P_3^1, P_3^3))$$

Exemples de fonctions récursives primitives (3)

$$\mathrm{PRED}(n) = \begin{cases} 0 & \text{Si } n = 0 \\ n-1 & \text{Sinon} \end{cases}$$

$$\mathrm{EQ}(n, m) = \begin{cases} 1 & \text{Si } n = m \\ 0 & \text{Sinon} \end{cases}$$

Exemples de fonctions récursives primitives (3)

$$\mathrm{PRED}(n) = \begin{cases} 0 & \text{Si } n = 0 \\ n - 1 & \text{Sinon} \end{cases}$$

$$\mathrm{EQ}(n, m) = \begin{cases} 1 & \text{Si } n = m \\ 0 & \text{Sinon} \end{cases}$$

$$\mathrm{PRED} = \mathrm{Prim}(0, P_2^2)$$

Exemples de fonctions récursives primitives (4)

$$J(x, y) = \frac{(x + y) \times (x + y + 1)}{2} + x$$

Est une bijection de $\mathbb{N}^2$ dans $\mathbb{N}$ qui est récursive primitive.
Les réciproques sont récursives primitives.

Exemples de fonctions récursives primitives (5)

Exponentielle, Factorielle, test de primalité, ...

Exemples de fonctions récursives primitives (5)

Exponentielle, Factorielle, test de primalité, ...

Une fonction calculable non récursive primitive ?

Hirarchie de Grzegorczyk

Hirarchie de Grzegorczyk

$\psi_{n+1}(m)$: itéré m fois de ψ_n sur $\psi_n(1)$.

Hirarchie de Grzegorczyk

$\psi_{n+1}(m)$: itéré m fois de ψ_n sur $\psi_n(1)$.

$$\begin{array}{rcl}\psi_0(m) & = & m+1 \\ \psi_{n+1}(0) & = & \psi_n(1) \\ \psi_{n+1}(m+1) & = & \psi_n(\psi_{n+1}(m))\end{array}$$

Pour tout n, ψ_n est une fonction récursive primitive.

Hirarchie de Grzegorczyk(2)

$$\psi_0(m) = m + 1$$
$$\psi_1(m) =$$

Hirarchie de Grzegorczyk(2)

$$\psi_0(m) = m + 1$$

$$\psi_1(m) = m + 2$$

Hirarchie de Grzegorczyk(2)

$$\psi_0(m) = m + 1$$
$$\psi_1(m) = m + 2$$

$$\psi_2(m) =$$

Hirarchie de Grzegorczyk(2)

$$\psi_0(m) = m + 1$$

$$\psi_1(m) = m + 2$$

$$\psi_2(m) = 2m + 3$$

Hirarchie de Grzegorczyk(2)

$$\psi_0(m) = m + 1$$

$$\psi_1(m) = m + 2$$

$$\psi_2(m) = 2m + 3$$

$$\psi_3(m) =$$

Hirarchie de Grzegorczyk(2)

$$\psi_0(m) = m + 1$$
$$\psi_1(m) = m + 2$$

$$\psi_2(m) = 2m + 3$$

$$\psi_3(m) = \alpha 2^m + \beta$$

Hirarchie de Grzegorczyk(2)

$$\psi_0(m) = m + 1$$

$$\psi_1(m) = m + 2$$

$$\psi_2(m) = 2m + 3$$

$$\psi_3(m) = \alpha 2^m + \beta$$

$\alpha = 8, \beta = -3$

Propriétés de la suite ψ_n

1. $\psi_n(m) \geq m + 1$
2. $\psi_n(m)$ est strictement croissante en n et en m
3. $\psi_k(\psi_m(n)) \leq \psi_{2+\max(k,m)}(n)$

Propriétés de la suite ψ_n

Montrons que $\psi_{n+1}(m) > \psi_n(m)$. (Supposant la croissance par rapport à m)

Propriétés de la suite ψ_n

Montrons que $\psi_{n+1}(m) > \psi_n(m)$. (Supposant la croissance par rapport à m)

Récurrence sur n puis récurrence sur m

Propriétés de la suite ψ_n

Montrons que $\psi_{n+1}(m) > \psi_n(m)$. (Supposant la croissance par rapport à m)

Récurrence sur n puis récurrence sur m

Si $n = 0$, $\psi_{n+1}(m) = m + 2 \geq \psi_0(m) + 1$.

Propriétés de la suite ψ_n

Montrons que $\psi_{n+1}(m) > \psi_n(m)$. (Supposant la croissance par rapport à m)

Récurrence sur n puis récurrence sur m

Si $n = 0$, $\psi_{n+1}(m) = m + 2 \geq \psi_0(m) + 1$.

$\psi_{n+2}(0) = \psi_{n+1}(1) \geq \psi_n(1) + 1$ (par hypothèse de récurrence). Or $\psi_n(1) = \psi_{n+1}(0)$. Donc $\psi_{n+2}(0) \geq \psi_{n+1}(0) + 1$.

Propriétés de la suite ψ_n

Montrons que $\psi_{n+1}(m) > \psi_n(m)$. (Supposant la croissance par rapport à m)

Récurrence sur n puis récurrence sur m

Si $n = 0$, $\psi_{n+1}(m) = m + 2 \geq \psi_0(m) + 1$.

$\psi_{n+2}(0) = \psi_{n+1}(1) \geq \psi_n(1) + 1$ (par hypothèse de récurrence). Or $\psi_n(1) = \psi_{n+1}(0)$. Donc $\psi_{n+2}(0) \geq \psi_{n+1}(0) + 1$.

$\psi_{n+2}(m+1) = \psi_{n+1}(\psi_{n+2}(m)) \geq \psi_{n+1}(\psi_{n+1}(m) + 1)$ (par hypothèse de récurrence et par croissance de ψ_{n+1}.)

Propriétés de la suite ψ_n

Montrons que $\psi_{n+1}(m) > \psi_n(m)$. (Supposant la croissance par rapport à m)

Récurrence sur n puis récurrence sur m

Si $n = 0$, $\psi_{n+1}(m) = m + 2 \geq \psi_0(m) + 1$.

$\psi_{n+2}(0) = \psi_{n+1}(1) \geq \psi_n(1) + 1$ (par hypothèse de récurrence). Or $\psi_n(1) = \psi_{n+1}(0)$. Donc $\psi_{n+2}(0) \geq \psi_{n+1}(0) + 1$.

$\psi_{n+2}(m+1) = \psi_{n+1}(\psi_{n+2}(m)) \geq \psi_{n+1}(\psi_{n+1}(m) + 1)$ (par hypothèse de récurrence et par croissance de ψ_{n+1}.)

$\psi_{n+1}(\psi_{n+1}(m) + 1) \geq \psi_n(\psi_{n+1}(m) + 1) + 1$ (par hypothèse de récurrence),

Propriétés de la suite ψ_n

Montrons que $\psi_{n+1}(m) > \psi_n(m)$. (Supposant la croissance par rapport à m)

Récurrence sur n puis récurrence sur m

Si $n = 0$, $\psi_{n+1}(m) = m + 2 \geq \psi_0(m) + 1$.

$\psi_{n+2}(0) = \psi_{n+1}(1) \geq \psi_n(1) + 1$ (par hypothèse de récurrence). Or $\psi_n(1) = \psi_{n+1}(0)$. Donc $\psi_{n+2}(0) \geq \psi_{n+1}(0) + 1$.

$\psi_{n+2}(m+1) = \psi_{n+1}(\psi_{n+2}(m)) \geq \psi_{n+1}(\psi_{n+1}(m) + 1)$ (par hypothèse de récurrence et par croissance de ψ_{n+1}.)

$\psi_{n+1}(\psi_{n+1}(m) + 1) \geq \psi_n(\psi_{n+1}(m) + 1) + 1$ (par hypothèse de récurrence),

et $\psi_n(\psi_{n+1}(m) + 1) + 1 \geq \psi_n(\psi_{n+1}(m)) + 1 = \psi_{n+1}(m+1) + 1$.

$\psi_k(\psi_m(n)) \leq \psi_{2+\max(k,m)}(n)$

Soit $M = \max(k, m)$.

$$\begin{array}{rcll} \psi_k(\psi_m(n)) & \leq & \psi_M(\psi_{M+1}(n)) & \\ & \leq & \psi_{M+1}(n+1) & \\ & \leq & \psi_{M+1}(\psi_1(n-1)) & \text{Si } n > 0 \\ & \leq & \psi_{M+1}(\psi_{M+2}(n-1)) & \\ & \leq & \psi_{M+2}(n) & \end{array}$$

et, si $n = 0$, $\psi_{M+1}(n+1) \leq \psi_{M+2}(0)$ et on a encore l'ingalit.

Récursion primitive et suite ψ_n

Toute fonction récursive primitive est majorée par une fonction de la hiérarchie de Grzegorczyk:

$$\forall \xi, \exists k_\xi \in \mathbb{N}.\ \forall \vec{n}.\ \xi(\vec{n}) \leq \psi_{k_\xi}(\max(\vec{n}))$$

Récursion primitive et suite ψ_n

Toute fonction récursive primitive est majorée par une fonction de la hiérarchie de Grzegorczyk:

$$\forall \xi, \exists k_\xi \in \mathbb{N}.\ \forall \vec{n}.\ \xi(\vec{n}) \leq \psi_{k_\xi}(\max(\vec{n}))$$

Vrai pour les fonctions de base

Récursion primitive et suite ψ_n

Toute fonction récursive primitive est majorée par une fonction de la hiérarchie de Grzegorczyk:

$$\forall \xi, \exists k_\xi \in \mathbb{N}.\ \forall \vec{n}.\ \xi(\vec{n}) \leq \psi_{k_\xi}(\max(\vec{n}))$$

Vrai pour les fonctions de base

Stable par composition.

Récursion primitive et suite ψ_n

Si $\theta(\vec{x}) \leq \psi_{k_\theta}(\max(\vec{x}))$ et $\xi(\vec{n}) \leq \psi_{k_\xi}(\max(\vec{n}))$

$$\phi(m, \vec{n}) = \begin{cases} \xi(\vec{n}) & \text{si } m = 0 \\ \theta(\phi(m-1, \vec{n}), m-1, \vec{n}) & \text{si } m > 0 \end{cases}$$

Récursion primitive et suite ψ_n

Si $\theta(\vec{x}) \leq \psi_{k_\theta}(\max(\vec{x}))$ et $\xi(\vec{n}) \leq \psi_{k_\xi}(\max(\vec{n}))$

$$\phi(m, \vec{n}) = \begin{cases} \xi(\vec{n}) & \text{si } m = 0 \\ \theta(\phi(m-1, \vec{n}), m-1, \vec{n}) & \text{si } m > 0 \end{cases}$$

$$\phi(m, \vec{n}) \quad \leq \quad \psi_{k_\theta}^{m}(\psi_{k_\xi}(\max(\vec{n})))$$

Récursion primitive et suite ψ_n

Si $\theta(\vec{x}) \leq \psi_{k_\theta}(\max(\vec{x}))$ et $\xi(\vec{n}) \leq \psi_{k_\xi}(\max(\vec{n}))$

$$\phi(m, \vec{n}) = \begin{cases} \xi(\vec{n}) & \text{si } m = 0 \\ \theta(\phi(m-1, \vec{n}), m-1, \vec{n}) & \text{si } m > 0 \end{cases}$$

$$\begin{aligned} \phi(m, \vec{n}) &\leq \psi_{k_\theta}^{m}(\psi_{k_\xi}(\max(\vec{n}))) \\ &\leq \psi_{1+k_\theta}(m + \psi_{k_\xi}(\max(\vec{n}))) \end{aligned}$$

La fonction d'Ackermann

$$Ack(n, m) = \psi_n(m)$$

La fonction d'Ackermann

$$Ack(n, m) = \psi_n(m)$$

Ack n'est pas récursive primitive.

La fonction d'Ackermann

$$Ack(n, m) = \psi_n(m)$$

Ack n'est pas récursive primitive.

Par l'absurde: si c'était le cas, il existerait m tel que, pour tout n,

$$Ack(n, n) \leq \psi_m(n)$$

La minimisation

$$\phi(\vec{n}) = \min\{m \mid \xi(m, \vec{n}) = 0\}$$

La minimisation

$$\phi(\vec{n}) = \min\{m \mid \xi(m, \vec{n}) = 0\}$$

Fonctions *partielles* de $\mathbb{N}^k$ dans $\mathbb{N}$

$$\phi(\vec{n}) = \min\{m \mid \xi(m, \vec{n}) = 0 \,\&\, \forall k < m. \xi(k, \vec{n}) \neq \bot\}$$

La minimisation

$$\phi(\vec{n}) = \min\{m \mid \xi(m, \vec{n}) = 0\}$$

Fonctions *partielles* de $\mathbb{N}^k$ dans $\mathbb{N}$

$$\phi(\vec{n}) = \min\{m \mid \xi(m, \vec{n}) = 0 \,\&\, \forall k < m. \xi(k, \vec{n}) \neq \bot\}$$

Fonctions récursives partielles: clôture par composition, récursion primitive, minimisation.

La minimisation

$$\phi(\vec{n}) = \min\{m \mid \xi(m, \vec{n}) = 0\}$$

Fonctions *partielles* de $\mathbb{N}^k$ dans $\mathbb{N}$

$$\phi(\vec{n}) = \min\{m \mid \xi(m, \vec{n}) = 0 \,\&\, \forall k < m. \xi(k, \vec{n}) \neq \perp\}$$

Fonctions récursives partielles: clôture par composition, récursion primitive, minimisation.

Fonctions utilisant des boucles while.

A l'appui de la thèse de Church

f récursive partielle

ssi

Il existe une machine de Turing M_f qui s'arrête exactement sur les entiers pour lesquels f est définie et $M_f(\vec{n}) = f(\vec{n})$.

Les fonctions récursives partielles sont Turing (semi-) calculables

Les fonctions de base sont Turing calculables

Les fonctions Turing (semi-) calculables sont closes par composition, récursion primitive et minimisation.

Cloture par minimisation

Soit $f(\overline{n}) = \min\{x \mid g(x, \overline{n}) = 0\}$.

M_g la machine qui calcule g.

Cloture par minimisation

Soit $f(\overline{n}) = \min\{x \mid g(x, \overline{n}) = 0\}$.

M_g la machine qui calcule g.

M_f a deux rubans, la donnée $\vec{n}$ est sur le ruban 2, 0 sur le ruban 1.

while $M_g(x_1, x_2) \neq 0$ do $x_1 + +$
return x_1

Les fonctions Turing calculables sont récursives partielles

- Configuration $(q, w_1, w_2) \quad \mapsto \quad C(q, w_1, w_2) \in \mathbb{N}$
  C injective

Les fonctions Turing calculables sont récursives partielles

- Configuration $(q, w_1, w_2) \quad \mapsto \quad C(q, w_1, w_2) \in \mathbb{N}$
  C injective

$$C(q, w_1, w_2) = J(C(q), J(C(\widetilde{w_1}), \times C(w_2)))$$

Les fonctions Turing calculables sont récursives partielles

- Configuration $(q, w_1, w_2) \quad \mapsto \quad C(q, w_1, w_2) \in \mathbb{N}$
  C injective

$$C(q, w_1, w_2) = J(C(q), J(C(\widetilde{w_1}), \times C(w_2)))$$

- $f_0 : \vec{n} \to C(q_0, \epsilon, \$\vec{n})$

  Est récursive primitive

Les fonctions Turing calculables sont récursives partielles

- Configuration $(q, w_1, w_2) \quad \mapsto \quad C(q, w_1, w_2) \in \mathbb{N}$
  C injective

$$C(q, w_1, w_2) = J(C(q), J(C(\widetilde{w_1}), \times C(w_2)))$$

- $f_0 : \vec{n} \to C(q_0, \epsilon, \$\vec{n})$

  Est récursive primitive
- $g_M : C(q, w_1, w_2) \to C(q', w_1', w_2')$ si
  $(q, w_1, w_2) \vdash_M (q', w_1', w_2')$

  Est récursive primitive

Les fonctions Turing calculables sont récursives partielles

- Configuration $(q, w_1, w_2) \quad \mapsto \quad C(q, w_1, w_2) \in \mathbb{N}$
  C injective

$$C(q, w_1, w_2) = J(C(q), J(C(\widetilde{w_1}), \times C(w_2)))$$

- $f_0 : \vec{n} \to C(q_0, \epsilon, \$\vec{n})$

  Est récursive primitive
- $g_M : C(q, w_1, w_2) \to C(q', w_1', w_2')$ si $(q, w_1, w_2) \vdash_M (q', w_1', w_2')$

  Est récursive primitive

  - $\mathrm{state}(C(q, w_1, w_2)) = q$ est récursive primitive

Les fonctions Turing calculables sont récursives partielles

- Configuration $(q, w_1, w_2) \quad \mapsto \quad C(q, w_1, w_2) \in \mathbb{N}$
  C injective

$$C(q, w_1, w_2) = J(C(q), J(C(\widetilde{w_1}), \times C(w_2)))$$

- $f_0 : \vec{n} \to C(q_0, \epsilon, \$\vec{n})$

  Est récursive primitive
- $g_M : C(q, w_1, w_2) \to C(q', w_1', w_2')$ si
  $(q, w_1, w_2) \vdash_M (q', w_1', w_2')$

  Est récursive primitive

  - state$(C(q, w_1, w_2)) = q$ est récursive primitive
  - first$(C(q, w_1, w_2))$ qui renvoie la première lettre de w_2 (ou B) est récursive primitive

Les fonctions Turing calculables sont récursives partielles

- Configuration $(q, w_1, w_2) \quad \mapsto \quad C(q, w_1, w_2) \in \mathbb{N}$
  C injective

$$C(q, w_1, w_2) = J(C(q), J(C(\widetilde{w_1}), \times C(w_2)))$$

- $f_0 : \vec{n} \to C(q_0, \epsilon, \$\vec{n})$

  Est récursive primitive
- $g_M : C(q, w_1, w_2) \to C(q', w_1', w_2')$ si $(q, w_1, w_2) \vdash_M (q', w_1', w_2')$

  Est récursive primitive

  - $\mathrm{state}(C(q, w_1, w_2)) = q$ est récursive primitive
  - $\mathrm{first}(C(q, w_1, w_2))$ qui renvoie la première lettre de w_2 (ou B) est récursive primitive
  - if $\mathrm{state}(n) = q \wedge \mathrm{first}(n) = a$ then $f_1(n)$ else $f_2(n)$ est récursive primitive

Les fonctions Turing calculables sont récursives partielles

- Configuration $(q, w_1, w_2) \quad \mapsto \quad C(q, w_1, w_2) \in \mathbb{N}$
  C injective

$$C(q, w_1, w_2) = J(C(q), J(C(\widetilde{w_1}), \times C(w_2)))$$

- $f_0 : \vec{n} \to C(q_0, \epsilon, \$\vec{n})$

  Est récursive primitive
- $g_M : C(q, w_1, w_2) \to C(q', w_1', w_2')$ si $(q, w_1, w_2) \vdash_M (q', w_1', w_2')$

  Est récursive primitive

  - $\text{state}(C(q, w_1, w_2)) = q$ est récursive primitive
  - $\text{first}(C(q, w_1, w_2))$ qui renvoie la première lettre de w_2 (ou B) est récursive primitive
  - if $\text{state}(n) = q \wedge \text{first}(n) = a$ then $f_1(n)$ else $f_2(n)$ est récursive primitive
  - $C(aw) \mapsto C(w)$ et $C(w) \mapsto C(aw)$ sont récursives primitives

Les fonctions Turing calculables sont récursives partielles

- $h_M(m, \vec{n}) = \begin{cases} f_0(\vec{n}) & \text{Si } m = 0 \\ g_M(h(m-1, \vec{n})) & \text{Sinon} \end{cases}$

  Est récursive primitive.

Les fonctions Turing calculables sont récursives partielles

- $h_M(m, \vec{n}) = \begin{cases} f_0(\vec{n}) & \text{Si } m = 0 \\ g_M(h(m-1, \vec{n})) & \text{Sinon} \end{cases}$

  Est récursive primitive.
  Calcul de la *m*ième configuration de la machine de Turing.

- f_1: $C(\gamma) \mapsto 0$ si γ est finale et $C(\gamma) \mapsto 1$ sinon

  Est récursive primitive

Les fonctions Turing calculables sont récursives partielles

- $h_M(m, \vec{n}) = \begin{cases} f_0(\vec{n}) & \text{Si } m = 0 \\ g_M(h(m-1, \vec{n})) & \text{Sinon} \end{cases}$

  Est récursive primitive.
  Calcul de la $m$ième configuration de la machine de Turing.
- f_1: $C(\gamma) \mapsto 0$ si γ est finale et $C(\gamma) \mapsto 1$ sinon

  Est récursive primitive
- $$t_M(\vec{n}) = \min\{m \mid f_1(h_M(m, \vec{n})) = 0\}$$

  temps de calcul de M sur $\vec{n}$

Les fonctions Turing calculables sont récursives partielles

- $h_M(m,\vec{n}) = \begin{cases} f_0(\vec{n}) & \text{Si } m = 0 \\ g_M(h(m-1,\vec{n})) & \text{Sinon} \end{cases}$

  Est récursive primitive.
  Calcul de la $m$ième configuration de la machine de Turing.
- f_1: $C(\gamma) \mapsto 0$ si γ est finale et $C(\gamma) \mapsto 1$ sinon

  Est récursive primitive
- $$t_M(\vec{n}) = \min\{m \mid f_1(h_M(m,\vec{n})) = 0\}$$

  temps de calcul de M sur $\vec{n}$
- $$f_M(\vec{n}) = \mathrm{clean}(h_M(t_M(\vec{n}),\vec{n}))$$

Théorème de Kleene

Une seule minimisation suffit

Une seule boucle while suffit

Élimination de la récursion primitive

$\mathcal{C}$ le plus petit ensemble des entiers dans les entiers tel que:

- Les fonctions initiales sont dans $\mathcal{C}$
- $+, \times, =$ sont dans $\mathcal{C}$
- $\mathcal{C}$ est clos par minimisation et composition

Fonctions dans $\mathcal{C}$

Conjonction de prédicats

$(P_1 \wedge P_2)(\overline{n}) =$

Fonctions dans $\mathcal{C}$

Conjonction de prédicats

$(P_1 \wedge P_2)(\overline{n}) = P_1(\overline{n}) \times P_2(\overline{n})$

Fonctions dans $\mathcal{C}$

Conjonction de prédicats

$(P_1 \wedge P_2)(\overline{n}) = P_1(\overline{n}) \times P_2(\overline{n})$

Négation de prédicats

$\neg P(\overline{n}) =$

Fonctions dans $\mathcal{C}$

Conjonction de prédicats

$(P_1 \wedge P_2)(\overline{n}) = P_1(\overline{n}) \times P_2(\overline{n})$

Négation de prédicats

$\neg P(\overline{n}) = P(\overline{n}) == 0$

Fonctions dans $\mathcal{C}$

Conjonction de prédicats

$(P_1 \wedge P_2)(\overline{n}) = P_1(\overline{n}) \times P_2(\overline{n})$

Négation de prédicats

$\neg P(\overline{n}) = P(\overline{n}) == 0$

Conditionelle

$$\text{if } P(\overline{n}) \text{ then } f(\overline{n}) \text{ else} g(\overline{n}) =$$

Fonctions dans $\mathcal{C}$

Conjonction de prédicats

$(P_1 \wedge P_2)(\overline{n}) = P_1(\overline{n}) \times P_2(\overline{n})$

Négation de prédicats

$\neg P(\overline{n}) = P(\overline{n}) == 0$

Conditionelle

$$\text{if } P(\overline{n}) \text{ then } f(\overline{n}) \text{ else} g(\overline{n}) = P(\overline{n}) \times f(\overline{n}) + \neg P(\overline{n}) \times g(\overline{n})$$

Fonctions dans $\mathcal{C}$

Conjonction de prédicats

$(P_1 \wedge P_2)(\overline{n}) = P_1(\overline{n}) \times P_2(\overline{n})$

Négation de prédicats

$\neg P(\overline{n}) = P(\overline{n}) == 0$

Conditionelle

$$\text{if } P(\overline{n}) \text{ then } f(\overline{n}) \text{ else} g(\overline{n}) = P(\overline{n}) \times f(\overline{n}) + \neg P(\overline{n}) \times g(\overline{n})$$

Soustraction entière

$$x \setminus y \quad =$$

Fonctions dans $\mathcal{C}$

Conjonction de prédicats

$(P_1 \wedge P_2)(\overline{n}) = P_1(\overline{n}) \times P_2(\overline{n})$

Négation de prédicats

$\neg P(\overline{n}) = P(\overline{n}) == 0$

Conditionelle

$$\text{if } P(\overline{n}) \text{ then } f(\overline{n}) \text{ else} g(\overline{n}) = P(\overline{n}) \times f(\overline{n}) + \neg P(\overline{n}) \times g(\overline{n})$$

Soustraction entière

$$x \setminus y \quad = \quad \max\{z \mid z + y == x \vee z == 0\}$$

Fonctions dans $\mathcal{C}$

Conjonction de prédicats

$(P_1 \wedge P_2)(\overline{n}) = P_1(\overline{n}) \times P_2(\overline{n})$

Négation de prédicats

$\neg P(\overline{n}) = P(\overline{n}) == 0$

Conditionelle

$$\text{if } P(\overline{n}) \text{ then } f(\overline{n}) \text{ else} g(\overline{n}) = P(\overline{n}) \times f(\overline{n}) + \neg P(\overline{n}) \times g(\overline{n})$$

Soustraction entière

$$\begin{aligned} x \setminus y &= \max\{z \mid z + y == x \vee z == 0\} \\ &= \begin{cases} 0 & \text{Si } \min\{z \mid z + y == x \vee z == x\} == x \\ \min\{z \mid z + y == x \vee z == x\} & \text{Sinon} \end{cases} \end{aligned}$$

Fonctions dans $\mathcal{C}$ (2)

Fonction J

$J(x, y) \stackrel{\text{def}}{=} \frac{(x+y)\times(x+y+1)}{2} + x =$

Fonctions dans $\mathcal{C}$ (2)

Fonction J

$J(x, y) \stackrel{\text{def}}{=} \frac{(x+y)\times(x+y+1)}{2} + x =$
$x + \min\{z \mid 2 \times z == (x + y) \times (x + y + 1)\}$

Fonctions dans $\mathcal{C}$ (2)

Fonction J

$J(x, y) \stackrel{\text{def}}{=} \frac{(x+y)\times(x+y+1)}{2} + x =$

$x + \min\{z \mid 2 \times z == (x+y) \times (x+y+1)\}$

Fonction K

$K(z) =$

Fonctions dans $\mathcal{C}$ (2)

Fonction J

$J(x, y) \stackrel{\text{def}}{=} \frac{(x+y)\times(x+y+1)}{2} + x =$
$x + \min\{z \mid 2 \times z == (x+y) \times (x+y+1)\}$

Fonction K

$K(z) = z \setminus (\frac{f(z)\times(f(z)-1)}{2})$ où
$f(z) = min\{x \mid x \times (x+1) \geq 2 \times z + 1\}$

Fonctions dans $\mathcal{C}$ (2)

Fonction J

$J(x,y) \stackrel{\text{def}}{=} \frac{(x+y)\times(x+y+1)}{2} + x =$
$x + \min\{z \mid 2 \times z == (x+y) \times (x+y+1)\}$

Fonction K

$K(z) = z \setminus (\frac{f(z)\times(f(z)-1)}{2})$ où
$f(z) = min\{x \mid x \times (x+1) \geq 2 \times z + 1\}$

Quantification bornée

$$Q(x) \stackrel{\text{def}}{=} \forall y < x.P(x,y) =$$

Fonctions dans $\mathcal{C}$ (2)

Fonction J

$J(x, y) \stackrel{\text{def}}{=} \frac{(x+y)\times(x+y+1)}{2} + x =$
$x + \min\{z \mid 2 \times z == (x+y) \times (x+y+1)\}$

Fonction K

$K(z) = z \setminus (\frac{f(z)\times(f(z)-1)}{2})$ où
$f(z) = min\{x \mid x \times (x+1) \geq 2 \times z + 1\}$

Quantification bornée

$$Q(x) \stackrel{\text{def}}{=} \forall y < x.P(x, y) = \quad \min\{z \mid \neg P(x, z) \vee z == x\} == x$$

Fonctions dans $\mathcal{C}$ (2)

Fonction J

$J(x,y) \stackrel{\text{def}}{=} \frac{(x+y)\times(x+y+1)}{2} + x =$
$x + \min\{z \mid 2 \times z == (x+y) \times (x+y+1)\}$

Fonction K

$K(z) = z \setminus (\frac{f(z)\times(f(z)-1)}{2})$ où
$f(z) = min\{x \mid x \times (x+1) \geq 2 \times z + 1\}$

Quantification bornée

$$Q(x) \stackrel{\text{def}}{=} \forall y < x . P(x,y) = \quad \min\{z \mid \neg P(x,z) \vee z == x\} == x$$

reste

$\texttt{reste}(x,y) =$

Fonctions dans $\mathcal{C}$ (2)

Fonction J

$J(x,y) \stackrel{\text{def}}{=} \frac{(x+y)\times(x+y+1)}{2} + x =$
$x + \min\{z \mid 2 \times z == (x+y) \times (x+y+1)\}$

Fonction K

$K(z) = z \setminus (\frac{f(z)\times(f(z)-1)}{2})$ où
$f(z) = min\{x \mid x \times (x+1) \geq 2 \times z + 1\}$

Quantification bornée

$$Q(x) \stackrel{\text{def}}{=} \forall y < x.P(x,y) = \quad \min\{z \mid \neg P(x,z) \vee z == x\} == x$$

reste

$\texttt{reste}(x,y) = \min\{z \mid \exists q \leq x.y \times q + z == x\}$

Fonction β de Gödel

$\beta \in \mathcal{C}$ tq

$$\forall n \in \mathbb{N}, \forall a_0, \dots, a_n \in \mathbb{N}, \exists d \in \mathbb{N}, \forall i \in \{0, \dots, n\}$$
$$\beta(d, i) = a_i$$

Fonction β de Gödel

$\beta \in \mathcal{C}$ tq

$$\forall n \in \mathbb{N}, \forall a_0, \ldots, a_n \in \mathbb{N}, \exists d \in \mathbb{N}, \forall i \in \{0, \ldots, n\}$$
$$\beta(d, i) = a_i$$

d encode la séquence $a_0, \ldots a_n$.

Définition de β

- $N = \max(n, a_0, \dots, a_n)$, $u_i = 1 + (i+1)N!$
- Lemme des restes chinois: $\{z = a_i \mod u_i \mid i = 0, \dots, n\}$ a une unique solution b telle que $d < u_0 \times \cdots \times u_n$
- $d = J(b, N!)$
- $\beta(x, y) = \texttt{reste}(K(x), 1 + (y+1) \times L(x))$

Définition de β

- $N = \max(n, a_0, \ldots, a_n),\ u_i = 1 + (i+1)N!$
- Lemme des restes chinois: $\{z = a_i \mod u_i \mid i = 0, \ldots, n\}$ a une unique solution b telle que $d < u_0 \times \cdots \times u_n$
- $d = J(b, N!)$
- $\beta(x, y) = \texttt{reste}(K(x), 1 + (y+1) \times L(x))$

$$\beta(d, i) \quad = \quad \texttt{reste}(K(d), 1 + (i+1)L(d))$$

Définition de β

- $N = \max(n, a_0, \ldots, a_n)$, $u_i = 1 + (i+1)N!$
- Lemme des restes chinois: $\{z = a_i \mod u_i \mid i = 0, \ldots, n\}$ a une unique solution b telle que $d < u_0 \times \cdots \times u_n$
- $d = J(b, N!)$
- $\beta(x, y) = \texttt{reste}(K(x), 1 + (y+1) \times L(x))$

$$\begin{aligned} \beta(d, i) &= \texttt{reste}(K(d), 1 + (i+1)L(d)) \\ &= \texttt{reste}(b, 1 + (i+1)N!) \end{aligned}$$

Définition de β

- $N = \max(n, a_0, \ldots, a_n)$, $u_i = 1 + (i+1)N!$
- Lemme des restes chinois: $\{z = a_i \mod u_i \mid i = 0, \ldots, n\}$ a une unique solution b telle que $d < u_0 \times \cdots \times u_n$
- $d = J(b, N!)$
- $\beta(x, y) = \texttt{reste}(K(x), 1 + (y+1) \times L(x))$

$$\begin{array}{rcl} \beta(d, i) & = & \texttt{reste}(K(d), 1 + (i+1)L(d)) \\ & = & \texttt{reste}(b, 1 + (i+1)N!) \\ & = & \texttt{reste}(b, u_i) \end{array}$$

Définition de β

- $N = \max(n, a_0, \ldots, a_n)$, $u_i = 1 + (i+1)N!$
- Lemme des restes chinois: $\{z = a_i \mod u_i \mid i = 0, \ldots, n\}$ a une unique solution b telle que $d < u_0 \times \cdots \times u_n$
- $d = J(b, N!)$
- $\beta(x, y) = \texttt{reste}(K(x), 1 + (y+1) \times L(x))$

$$\begin{aligned}
\beta(d, i) &= \texttt{reste}(K(d), 1 + (i+1)L(d)) \\
&= \texttt{reste}(b, 1 + (i+1)N!) \\
&= \texttt{reste}(b, u_i) \\
&= a_i
\end{aligned}$$

Théorème principal

$\mathcal{C}$ est l'ensemble des fonctions récursives partielles.

Preuve du théorème

Soit

$$f(x, \overline{n}) = \begin{cases} \xi(\overline{n}) & \text{Si } x = 0 \\ \psi(f(x-1, \overline{n}), x-1, \overline{n}) & \text{Sinon} \end{cases}$$

On suppose $\xi, \psi \in \mathcal{C}$.

Preuve du théorème

Soit

$$f(x,\overline{n}) = \begin{cases} \xi(\overline{n}) & \text{Si } x = 0 \\ \psi(f(x-1,\overline{n}), x-1, \overline{n}) & \text{Sinon} \end{cases}$$

On suppose $\xi, \psi \in \mathcal{C}$.

$S(x, d, \overline{n})$ exprime que d code la séquence

$$\xi(\overline{n}), \psi(\xi(\overline{n}), 0, \overline{n}), ..., \psi(f(x-1,\overline{n}), x-1, \overline{n})$$

Preuve du théorème

Soit

$$f(x,\overline{n}) = \begin{cases} \xi(\overline{n}) & \text{Si } x = 0 \\ \psi(f(x-1,\overline{n}), x-1, \overline{n}) & \text{Sinon} \end{cases}$$

On suppose $\xi, \psi \in \mathcal{C}$.

$S(x, d, \overline{n})$ exprime que d code la séquence

$$\xi(\overline{n}), \psi(\xi(\overline{n}), 0, \overline{n}), ..., \psi(f(x-1,\overline{n}), x-1, \overline{n})$$

$$S(x, d, \overline{n}) \overset{\text{def}}{=} \beta(d, 0) == \xi(\overline{n}) \wedge \forall i < x. \beta(d, i+1) == \psi(\beta(d, i), i, \overline{n})$$

Preuve du théorème

Soit

$$f(x,\overline{n}) = \begin{cases} \xi(\overline{n}) & \text{Si } x = 0 \\ \psi(f(x-1,\overline{n}), x-1, \overline{n}) & \text{Sinon} \end{cases}$$

On suppose $\xi, \psi \in \mathcal{C}$.

$S(x, d, \overline{n})$ exprime que d code la séquence

$$\xi(\overline{n}), \psi(\xi(\overline{n}), 0, \overline{n}), ..., \psi(f(x-1,\overline{n}), x-1, \overline{n})$$

$$S(x, d, \overline{n}) \stackrel{\text{def}}{=} \beta(d, 0) == \xi(\overline{n}) \wedge \forall i < x. \beta(d, i+1) == \psi(\beta(d, i), i, \overline{n})$$

$$f(x, \vec{n}) = \beta(\min\{d \mid S(x, d, \overline{n})\}, x)$$