HAI713I

Salle de TD : 36-410

Mot de passe Moodle : Turing

Plan du cours :

- Calculabilité
- Logique
- Complexité

MCC : max(exam, 0.7 exam + 0.3 CCs)

Cours 1: Intro

• Calculabilité

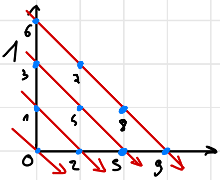
Def : ce qui est possible de calculer avec un ordinateur

Est-ce qu'on est capable d'obtenir une sortie, arrêt ? le temps d'exec. n'importe pas

2 types d'entrées : bijection des entiers dans les mots avec $\Sigma = \{0;1\}$ binaire + 1

- mots Σ^* $\quad \langle x, \langle y, z \rangle \rangle = \langle x, y, z \rangle$
- entiers $\mathbb{N}$ $\quad$ bijection de $(\mathbb{N}, \mathbb{N}) \longrightarrow \mathbb{N}$ $\quad \langle a, b \rangle \rightsquigarrow n$

ordre militaire: taille, puis alphabet



Une fonction est **calculable** ssi $\exists$ un programme qui la calcule

Fonction PPR : fonction partielle partiellement récursive

Une fonction est **totale** ssi elle est définie sur $\mathbb{N}$

$A \subset \mathbb{N}$ est **récursif** ssi $\left(\begin{array}{lcl} \mathbb{N} & \rightarrow & \{0;1\} \\ x & \rightarrow & \begin{cases} 1 & \text{si } x \in A \\ 0 & \text{sinon} \end{cases} \end{array} \right)$ est récursive totale

fonction caractéristique

Un ensemble est **énumérable** ssi $\exists$ un programme dont l'ensemble de convergeance est cet ensemble

Récursif $\Rightarrow$ Enumérable

Théorème de Post : si A énumérable && $\bar{A}$ énumérable alors A est récursif

Preuve : un programme qui dit si $x \in A$ ou non. Soit A tq x converge, $x \in A$

Soit B tq x converge ssi $x \in \bar{A}$

On utilise les pas. Le prog prend en paramètre t et donne 0 si a sur x ne termine pas en t étapes et donne $[a|x]+1$ sinon

```
def :
    t = 0
    while  step <a, x, t> = step <b, x, t> = 0
        t = t + 1
    if step <a, x, t> ≠ 0 :
        return 1
    else :
        return 0
```

$[a|x]$: résultat du programme a sur l'entrée x (diverge ↑ si erreur, converge ↓ sinon)

Universalité : $\exists u, \forall x,y \; [u \mid \langle x,y \rangle] = [x \mid y]$

$$\text{step}\langle x,y,t\rangle = \begin{cases} 0 & \text{si pas encore de convergeance} \\ [x\mid y]+1 & \text{sinon} \end{cases}$$

Cours 2 : l'arrêt

Problème de décision = Entrées → oui/non

ici Entrée : $\langle x,y \rangle$ Question : $[x \mid y]\downarrow$

Autre formulation : $f : \mathbb{N} \to \{0,1\}$, $\langle x,y\rangle \to \begin{cases} 1 & \text{si } [x\mid y]\downarrow \\ 0 & \text{sinon} \end{cases}$ **f pas calculable**

Autre formulation : $A = \{\langle x,y\rangle, [x \mid y]\downarrow\}$ $A \subset \mathbb{N}$ **A pas récursif**

Théorème de l'arrêt : il n'y a pas de programme qui résout le problème de l'arrêt

Preuve $f : \mathbb{N} \to \{0,1\}$

$$\langle x,y\rangle \to \begin{cases} 1 & \text{si } [x\mid y]\downarrow \\ 0 & \text{sinon} \end{cases}$$

On suppose que f est calculable

$\exists a \quad [a \mid \cdot] = f$

$b : x \longrightarrow$ if $[a \mid \langle x,x\rangle] = 0$ then return 0

else $\perp$ ↪ plante

$[b \mid b]$?

- si $[b \mid b]\downarrow$ alors $[a \mid \langle b,b\rangle] = 0$ alors $[b \mid b]\nearrow$
- si $[b \mid b]\uparrow$ alors $[a \mid \langle b,b\rangle] = 1$ alors $[b \mid b]\downarrow$

Contradiction

Donc f non calculable

Si on regarde la définition, on a montré que $K = \{x, [x \mid x]\downarrow\}$ n'est pas récursif

Le problème de l'arrêt est **indécidable**

K est **énumérable**

$c : x \longrightarrow$ if $[x \mid x]\downarrow$ then return 1

$W_c = \text{dom}\,[c \mid \cdot] = K$

* K n'est pas récursif
* $\overline{K}$ n'est pas énumérable

– Problème de l'arrêt sur 0

entrée : x question : $[x \mid 0]\downarrow$?

$A = \{x, [x \mid 0]\downarrow\}$

$c : \langle x,y\rangle \longrightarrow$ if $[x \mid x]\downarrow$ then return 0

$c_x : y \to$ if $[x \mid x]\downarrow$ then return 0

$x \to c_x$: " if $[x \mid x]\downarrow$ then return 0 "

↳ programme récursif total

$\forall x, \; x \in K \iff c_x \in A$ **Réduction**

Théorème SNM (données $\longrightarrow$ programmes)

$\exists$ une fonction récursive totale S^m_n

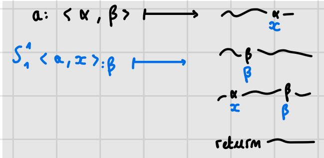
$\forall x, y \; [a \mid \langle x_1 \ldots x_m, y_1 \ldots y_n \rangle] = [S^m_n \langle x_1 \ldots x_m, y_1 \ldots y_n \rangle]$

$c : \langle x, y \rangle \longrightarrow$ if $[x \mid x]\downarrow$ then return 0

$S^1_1 \langle c, x \rangle : y \longrightarrow$ if $[x \mid x]\downarrow$ then return

$x \longrightarrow c_x$

$x \longrightarrow S^1_1 \langle c, x \rangle$



$x \in K \iff S^1_1 \langle c, x \rangle \in A$

- si $\exists\, S^1_1$ récursive totale alors $\forall\, m, n$ S^m_n récursive totale
- On a une composition récursive si et seulement si on a un théorème SNM

a, b $\exists\, c$ récursive totale $\forall x, y$ $[c \langle a, b \rangle \mid x] = [a \mid [b \mid x]]$

$= [a \mid \cdot] \circ [b \mid \cdot](x)$

Universel (programmes $\longrightarrow$ données)

$\exists\, u, \; [u \mid \langle x, y \rangle] = [x \mid y]$

Cours 3: Réduction Many-One

$\neq$ réduction Turing

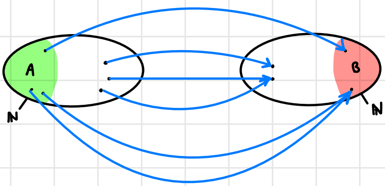
$\neq$ réduction One-One

$A, B \subset \mathbb{N}$

$A \leq_m B$ si $\exists\, f$ récursive totale

$\forall x, \; x \in A \iff f(x) \in B$

B pas plus compliqué que A



Prop. $\leq_m$ est un pré ordre

$\forall A \subset \mathbb{N} \qquad A \leq_m A$

$\forall A, B, C \qquad A \leq_m B$ et $B \leq_m C \Rightarrow A \leq_m C$

$g \circ f$ est une fonction de réduction pour $A \leq_m C$

Exemple: $K = \{x, [x \mid x]\downarrow\}$

$K_0 = \{x, [x \mid 0]\downarrow\}$

$b : \langle x, y \rangle \longmapsto$ si $[x \mid x]\downarrow$ alors return 0

sinon $\perp$

$S^1_1 \langle b, x \rangle : y \longmapsto$ si $[x \mid x]\downarrow$ alors return 0

$f : x \longmapsto S^1_1 \langle b, x \rangle$ montrons que f est une fonction de réduction de $K \longrightarrow K_0$

- f est récursive totale
- $x \in K \Rightarrow f(x) \in K_0$

si $x \in K$, alors $[f(x) \mid \cdot] = 0$ $(\forall y, \; [f(x) \mid y] = 0)$

donc $[f(x) \mid f(x)]\downarrow$, $f(x) \in K_0$

- si $x \notin K$ alors $[f(x) \mid \cdot] = \perp$

donc $[f(x) \mid f(x)]\uparrow$, $f(x) \notin K_0$

Théorème de Rice

trivial : $= \emptyset$ ou $= \mathbb{N}$

non trivial $\Rightarrow$ non récursif

Décidable $\Rightarrow$ Enumérable

Soit $\mathcal{C}$ une propriété sur les fonctions de $\mathbb{N}$ dans $\mathbb{N}$ (partielles ou totales)

Soit $P_{\mathcal{C}} = \{x, [x|\cdot] \in \mathcal{C}\}$

Alors soit $P_{\mathcal{C}}$ est trivial $P_{\mathcal{C}} = \emptyset$ ou $P_{\mathcal{C}} = \mathbb{N}$

soit $P_{\mathcal{C}}$ n'est pas récursif (= indécidable)

Preuve : si $\perp \in \mathcal{C}$ alors je prends $\overline{\mathcal{C}}$ comme propriété $\quad P_{\overline{\mathcal{C}}} = \overline{P_{\mathcal{C}}}$

d'où $\perp \notin \mathcal{C}$

Si $P_{\mathcal{C}}$ non trivial, soit $A \in P_{\mathcal{C}}$ (il existe car $P_{\mathcal{C}}$ pas trivial)

$$b : x, y \longmapsto \text{si } [x|x]\downarrow \text{ alors } [x|y]$$

$$[S_1^1 \langle b, x\rangle | \cdot] = \begin{cases} \text{si } x \in K \text{ alors } [b|\cdot] \\ \text{sinon } \perp \end{cases}$$

$$f : x \longmapsto S_1^1\langle b, x\rangle \text{ récursive totale}$$

$$\forall x,\ x \in K \Leftrightarrow f(x) \in P_{\mathcal{C}}$$

$K <_m P_{\mathcal{C}}$, donc $P_{\mathcal{C}}$ n'est pas récursif

Remarques :
- $\forall A$, $K <_m A$ alors A n'est pas récursif
- Soit p tq $[p|\cdot] = \perp$

  si $p \notin P_{\mathcal{C}}$ alors soit $P_{\mathcal{C}}$ vide, soit $K <_m P_{\mathcal{C}}$

  si $p \in P_{\mathcal{C}}$ alors soit $\overline{P_{\mathcal{C}}} = \emptyset$ (ie. $P_{\mathcal{C}} = \mathbb{N}$), soit $K <_m \overline{P_{\mathcal{C}}}$
- si $[a|\cdot] = [b|\cdot]$ alors soit $a \in P_{\mathcal{C}}$ et $b \in P_{\mathcal{C}}$

  soit $a \notin P_{\mathcal{C}}$ et $b \in P_{\mathcal{C}}$
- $B_a = \{x, [x|a]\downarrow\}$, $\mathcal{C} = \{f, a \in \text{dom} f\}$, $P_{\mathcal{C}} = B_a$

  $P_{\mathcal{C}}$ est non trivial et donc non récursif (Rice) $\quad B_a$ n'est pas récursif

Propriété : Si B est récursif, non trivial et $A < B$ alors A récursif

Preuve : $f : x \longrightarrow$ if $x \in A$ then return b $(b \in B)$

else return c $(c \notin B)$

Propriété : Si B énumérable, non trivial et $A < B$ alors A énumérable

Preuve : $B = W_b = \{x, [b|x]\downarrow\}$

f récursive totale

$a : x \longrightarrow$ return $[b|f(x)]$ $\qquad A = W_a$

Propriété : si A récursif et B énumérable non trivial, alors $A < B$

Preuve : $x \longrightarrow$ si $x \in A$ return b

sinon return c

Corolaire : si A est récursif et B non trivial alors $A < B$

Propriété : si A est énumérable alors $A < \mathbb{K}$

Preuve : $A = W_a$

$b : x, y \longrightarrow$ if $[a|x]\downarrow$ then return 0

Remarque: si $x \in A$ alors $\forall y\ [b \mid \langle x,y\rangle] = 0$ $\begin{cases} \nearrow \in \mathbb{K} \\ \searrow \notin \mathbb{K} \end{cases}$

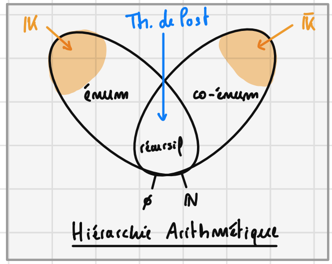
$x \notin A$

$f : x \longrightarrow S_1^1 \langle b, x\rangle$ récursive totale

$f(x) \in \mathbb{K} \iff x \in A$

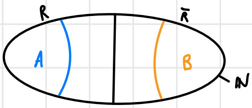
Rappel: $A < B \iff \bar{A} < \bar{B}$

$\boxed{A \text{ est co-blabla ssi } \bar{A} \text{ est blabla}}$

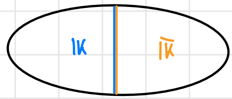


Cours 4: Inséparabilité

Def: A et B disjoints sont inséparables (récursivement) ssi $\nexists$ R récursif tq $A \subset R$ et $B \subset \bar{R}$



Remarque : $\mathbb{K}$ et $\bar{\mathbb{K}}$ sont inséparables



Théorème: il existe deux ensembles disjoints énumérables qui sont récursivement inséparables

$A = \{x,\ [x \mid x] = 0\}$ $\qquad B = \{x,\ [x \mid x] = 1\}$

Supposons qu' $\exists R$ récursif.

$A \subset R$ et $B \subset \bar{R}$

$\varphi_R : x \longrightarrow \begin{cases} 1 & \text{si } x \in R \\ 0 & \text{sinon} \end{cases}$

$\exists r\ \ \varphi_R = [r \mid \cdot\,]$

$\left.\begin{array}{l} \text{si } [r \mid r] = 1 \ ,\ r \in B \text{ mais } r \in R \\ \text{si } [r \mid r] = 0 \ ,\ r \in A \text{ mais } r \notin R \end{array}\right)$ contradiction : R n'existe pas

Théorème: $\exists$ fonction calculable partielle qui ne peut pas être étendue à une fonction calculable totale.

Preuve: $x \longrightarrow$ if $[x \mid x] = 0$ return 0

else if $[x \mid x] = 1$ return 1

else $\perp$

si on pouvait étendre ce programme

f calculable totale, si $x \in A$ alors $f(x) = 0$

si $x \in B$ alors $f(x) = 1$

b: $x \longrightarrow$ if $x = 0$ return 0

if $x = 1$ return 1

else return 1

$c : x \longrightarrow [b \mid f(x)]$ c calcule une fonction caractéristique d'un ensemble R récursif et sépare A et B contradiction

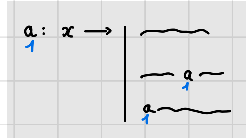
Retour sur la diagonalisation

Théorème de lycée (Cantor) : $\nexists$ de fonction surjective de $\mathbb{N}$ dans $\mathbb{R}$

un réel est une suite de $\mathbb{N} \to \{0,1\}$

$$\begin{matrix} r_{0,0} & r_{0,1} & r_{0,2} \\ r_{1,0} & r_{1,1} & r_{1,2} \\ r_{2,0} & r_{2,1} & r_{2,2} \end{matrix}$$

Cours 5 : Théorèmes de points fixes



nous autorise à utiliser a dans a

Théorème de la récursion Kleene : Soit f une fonction calculable totale, alors :
$\exists n \; [f(n) \mid \cdot] = [n \mid \cdot]$

Preuve : g récursive totale tq. $[g(x) \mid \cdot] = [[x \mid x] \mid \cdot]$

g existe par S^m_n, $b : x, y \longrightarrow [[x \mid x] \mid y] \Rightarrow g(x) = S^1_1 \langle b, x \rangle$
$\hookrightarrow = [u \mid \langle [x \mid x], y \rangle]$

m un programme qui calcule $f \circ g$

Soit $n = g(m)$, $[n \mid \cdot] = [g(m) \mid \cdot] = [[m \mid m] \mid \cdot] = [f \circ g(m) \mid \cdot] = [f(n) \mid \cdot]$
$\hookrightarrow [f(n) \mid \cdot] = [n \mid \cdot]$

Application : $\exists$ un prog. autoreproducteur $= \exists n, [n \mid \cdot] = n$

$f(a) : y \longrightarrow$ return (a)

existe par S^m_n, $b : x, y \longrightarrow$ return (a) $\quad f(a) = S^1_1 \langle b, a \rangle$

Exemple (C) :

```c
main() {
    char* b = "main(){char* b = %c %s %c; printf(b,34,b,34);}";
    printf(b,34,b,34);
}
```

Corrolaire : $\exists$ un programme n dont le domaine est $\mathbb{N} \setminus \{n\}$

Preuve : $x, y \longrightarrow 0$ si $x \neq y$
$\perp$ sinon

$f(x) = S^1_1 \langle a, x \rangle$

point fixe $[f(n) \mid y] = [n \mid y] = \begin{cases} 0 & \text{si } x \neq y \\ \perp & \text{sinon} \end{cases}$

<u>Nouvelle preuve Rice</u> : $a \in P_\varepsilon$, $b \notin P_\varepsilon$ $\quad f(x) = b$ si $x \in P_\varepsilon$, a sinon

Si P décidable alors f est récursive totale

point fixe si $n \in P_\varepsilon$ alors b
si $n \notin P_\varepsilon$ alors a $\Big\}$ contradiction

Améliorations (Roger) : ⭐ Fonctions partielles

Soit f calculable (~~totale~~), $\exists n \; [f(n) \mid \cdot] = [n \mid \cdot]$

$[g(x) \mid \cdot] = [[x \mid x] \mid \cdot]$, m calcule $f \circ g$, $n = g(m)$

$[n \mid \cdot] = [g(m) \mid \cdot] = [[m \mid m] \mid \cdot] = [f \circ g(m) \mid \cdot] = [f(n) \mid \cdot]$

$\hookrightarrow$ la preuve reste la même, pas besoin de f totale

⭐ Effectivité du point fixe

Si on donne un programme qui calcule f alors on a un programme qui calcule $f \circ g$

On applique g et on obtient n

Théorème : $\exists$ une fonction récursive totale n $\quad [[x \mid n(a)] \mid \cdot] = [n(a) \mid \cdot]$

Ensembles de points fixes

$\Pi_f = \{n, [n|\cdot] = [f(n)|\cdot]\}$

Propriété : Π_f est infini

Preuve : si Π_f fini, $\Pi_f = \{a_1, a_2, \dots, a_n\}$

$\exists b$ tq $\forall_i\ [b|\cdot] \neq [a_i|\cdot]$

$f' : x \longrightarrow$ si $x \in \Pi_f$ alors return b sinon return $f(x)$

f' calculable donc point fixe de f' : n

$[n|\cdot] = [f'(n)|\cdot]$ $n \neq a_1$ car b calcule autre chose que a_1

$\vdots$

$n \notin \Pi_f$

et $[n|\cdot] = [f'(n)|\cdot] = [f(n)|\cdot]$ contradiction

Propriété : $\exists$ des ensembles de points fixes non récursifs

Preuve : $f : x \longrightarrow a$

point fixe $\exists n$ $[n|\cdot] = [f(n)|\cdot] = [a|\cdot]$

Π_f pas récursif (Rice)

Il existe Π_f récursif non trivial

soit $R = \{r_0, r_1, \dots, r_n, \dots\}$ tq $[r_i|\cdot]$ tous différents

$f(x) = \begin{cases} \text{si } x = r_i \text{ alors } r_{i+1} \\ \text{sinon } r \end{cases}$ $\Pi_f = \mathbb{N} \setminus R$