Notions Préliminaires

- Langages formels
  - Alphabet, mot, langage
- Induction structurelle
  - Définition d’un langage par induction
  - Définition par induction d’une fonction sur les mots d’un langage
  - Preuve par induction d’une propriété sur les mots d’un langage

Langages Formels

- Alphabet
  - Ensemble de symboles utilisables *$A=\{a,b,c\}$*
- Mot ou expression
  - Une suite d'élément de l'alphabet *aaba*
- Longueur d’un mot
  - Nombre de symboles qui le composent *long(aaba)=4*
  - Le mot vide de longueur 0 est noté ε
- Concaténation notée **.**
  - Opération binaire associative sur les mots qui désigne le mot obtenu en mettant bout à bout les deux opérandes

    si m1 = ab et m2=bba, m1.m2 = abbba

Langages Formels

- Langage
  - Soit A un alphabet, un langage **L** sur A est un sous-ensemble des mots que l'on peut construire sur A
  - Exemples avec *$A=\{a,b,c\}$*

    $L_1 = \{aba, ba, cabb\}$

    $L_2 = \{\varepsilon, ab, aabb, aaabbb\ldots\}$ *les mots ayant autant de a que de b, les a précédents les b*

- Soit A un alphabet, **A*** désigne le langage de tous les mots que l'on peut construire sur A

  $A^*=\{\varepsilon, a, b, c, aa, ab, ac, ba, bb\ldots cccabccaaa,\ldots\}$

  Tout langage sur A est un sous-ensemble de A (et vice-versa)*

Langages Formels

- Outils de définition d'un langage
  - expression régulière, automate, grammaire, **définition par induction structurelle**
- Règles de construction *r* (règles de production)
  - Des fonctions partiellement définies sur A*
    - Données : soit n éléments $m_1, \ldots, m_n$ de A* (vérifiant parfois des conditions spécifiques). *n* est appelée l'arité de la fonction
    - Résultat : un nouvel élément de A*, noté $r(m_1, \ldots, m_n)$, défini à partir des données, de A et de l'opérateur de concaténation

Ex. r_1: $A^* \longrightarrow A^*$ $\quad$ r_2: $A^* \times A^* \longrightarrow A^*$

$m \longmapsto r_1(m) = a.m.b$ $\quad$ $m_1, m_2 \longmapsto r_2(m_1, m_2) = m_1.b.m_2$

Définition de langages par induction structurelle

- Définir un langage L par induction structurelle consiste à
  - Donner un alphabet A
  - Donner un sous-ensemble B de A* appelé la **base**
  - Donner un ensemble R de règles de **cons**truction sur A*

- L est alors défini (par induction) comme le plus petit ensemble $L \subseteq A^*$ tel que :

  (**base**) L contient B

  (**cons**) pour toute $r \in R$ et tout $m_1,\ldots m_n \in L$ (où n est l'arité de r), si $r(m_1,\ldots m_n)$ est défini alors $r(m_1,\ldots m_n) \in L$

Exemple

- Définition d'un langage Lp
  - Alphabet A={1,p,e}
  - (base) {pe}
  - (cons) Soit m un mot de Lp
    - (r_1) 1.m.1 est un mot de Lp
    - (r_2) si m contient un e (c'est-à-dire m=m'.e.m'')
      alors m'.1.e.1.m'' est un mot de Lp

$r_1 : A^* \longrightarrow A^*$

$m \longmapsto r_1(m)=1.m.1$

$r_2 : A^* \longrightarrow A^*$ définie uniquement pour les mots contenant au moins un e

$m=m'.e.m'' \longmapsto r_2(m)=m'.1.e.1.m''$

- Ainsi Lp={pe, 1pe1, p1e1, 11pe11, 1p1e11…}

Définition inductive d'une fonction sur les mots d'un langage défini par induction

- Soit L un langage défini par induction avec (B,R)

On définit une fonction f sur les mots de L par induction ainsi :

(base) pour tout mot m de B, **on fixe la valeur f(m)**

(cons) et pour tout $r \in R$ et tout $m_1, \ldots m_n \in L$ (si $r(m_1, \ldots m_n)$ est défini),
on définit $f(r(m_1, \ldots m_n))$ en fonction de $f(m_1), \ldots f(m_n)$

Exemple

- Définition d'une fonction donnant le nombre de 1 d'un mot de Lp

  nbr1(1pe1)=2, nbr1(1p1e11)=4

- Par induction :

nbr1 : Lp $\longrightarrow \mathbb{N}$

m $\longmapsto \mathrm{nbr1(m)}$

(base) si m=pe alors nbr1(pe) = 0

(cons)

- si m=r1(m')=1m'1 alors nbr1(m) = 2 + nbr1(m')
- si m=r2(m'em'')=m'1e1m''

  alors nbre1(m)=2 + nbr1(m'em'')

Preuve par induction structurelle

- Soit L un langage défini par induction avec (B,R) et P une propriété :

Prouver que tous les mots de L vérifient P par induction structurelle c'est :

(base) pour tout mot m de B**, prouver que m vérifie P**

(cons) et pour tout r de R et tout $m_1,\ldots m_n \in$ L (si $r(m_1,\ldots m_n)$ est défini), **prouver que** :

si $m_1,\ldots m_n$ vérifient P alors $r(m_1,\ldots m_n)$ vérifie P

Exemple

- Prouvons par induction structurelle que les mots du langage Lp « possède un nombre pair de 1 »

(base) le seul mot de la base est pe

nbr1(pe)=0 et 0 est pair

(cons)

- si m=r1(m')=1m'1

  par *hypothèse d'induction*, nbr1(m') est pair

  donc il existe k $\in \mathbb{N}$ tel que nbr1(m')=2.k

  or nbr1(m) = 2+nbr1(m') = 2.(1+k) qui est donc pair

- si m=r2(m'em'')=m'1e1m''

à terminer …