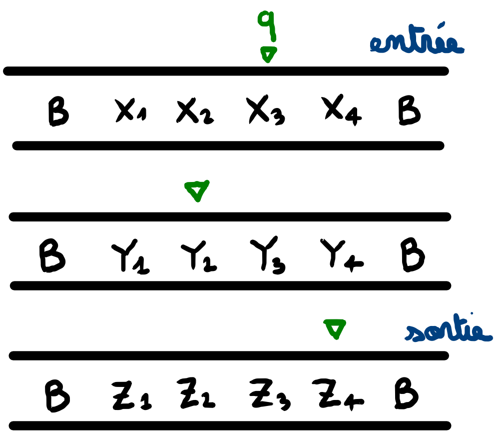
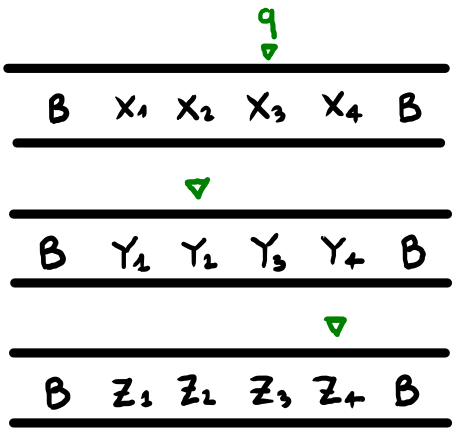
# Machines à rubans multiples

1 ruban d'entrée, 1 de sortie, 0 ou plus de travail




1 tête par ruban

les transitions lisent tous les rubans et en modifient un seul




la configuration est notée

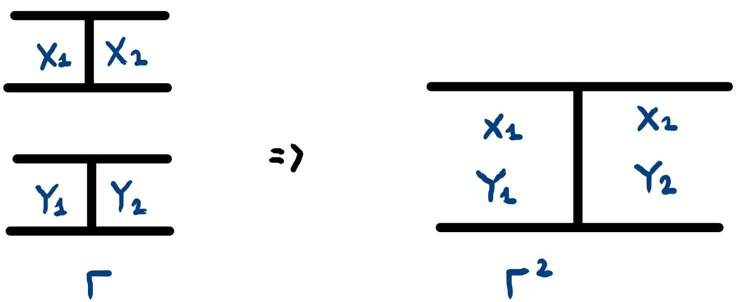
$$(X_1 X_2\, q\, X_3 X_4,\; Y_1 \triangleright Y_2 Y_3 Y_4,\; Z_1 Z_2 Z_3 \triangleright Z_4)$$

l'entrée est écrite sur le ruban d'entrée
la sortie de sortie

arrêt, acceptation, rejet : idem MT normale

# équivalence avec les MT simples

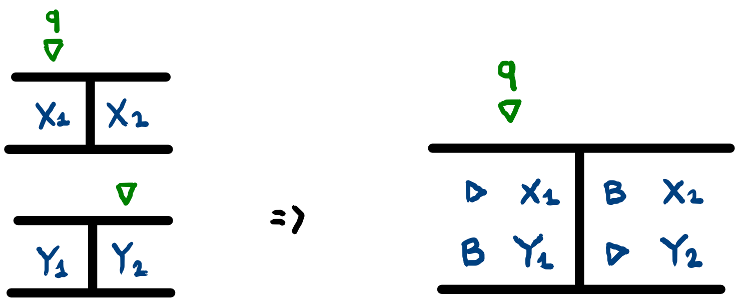
idée : stocker $n$ cases en une




plusieurs têtes : comment lire différentes cases en même temps ?

et les têtes ?

prévoir aussi $n$ sous-cases pour les têtes




$\Gamma$ $\qquad\qquad$ $\Gamma^2 \times \{\triangleright, B\}^2$

$2n$ sous-cases

## comment lire les rubans ?

- parcourir le ruban en stockant le contenu des cases pointées dans les états
- comment savoir où sont les têtes ? e.g. j'ai vu $n-1$ têtes, faut-il aller à droite ou à gauche pour voir la dernière ?

solution : mémoriser dans le registre le nombre de têtes à droite et à gauche de la position courante.

une fois toutes les têtes vues, choisir la transition.

# les MT en tant que programmes

comment sauvegarder une MT en binaire ?

il faut mémoriser

$Q$ voir $|Q|$

$q_a$ son n°

$\delta$ pour chaque transition $q \xrightarrow{X,\, Y,\, m} q'$

$q$ son n°

$\left.\begin{array}{l} X \\ Y \\ m \end{array}\right\}$ comment ?

$q'$ son n°

comment séparer ces informations ?

en les écrivant en unaire $n \rightarrow 0^n$

$Q$ # $q_a$ # $q_0 \xrightarrow{1,1,S} q_1$

séparateur

2 # 2 # $q_0$ # 1 # 1 # S # $q_1$ ...

00 1 00 1 0 1 00 1 00 1 000 1 00 ...

| | | | |
|---|---|---|---|
| | 0 | 0 | L |
| convention | 00 | 1 | R |
| | 000 | B | S |

on note $\langle M \rangle$ le code binaire $\in \{0,1\}^*$

de la MT $M$

$\mathcal{M}_m$ est la MT de code $m$

si $m$ n'est pas interprétable comme MT

$\mathcal{M}_m$ rejette tout, tout de suite

# complexité de Kolmogorov

$K(x)$ taille $|\langle M \rangle|$ de la plus petite MT $M$

produisant la sortie $x$ sur l'entrée vide

$(\text{bonjour})^{10^6}$ plus simple à écrire qu'un

mot aléatoire de longueur $7 \cdot 10^6$.

## interpréter le code d'une MT

existe t-il un interpréteur de code ?

ie $\mathcal{U}$ machine universelle tq $\mathcal{U}$ simule

$M(x)$ sur l'entrée $\langle M \rangle \# x$ , $M$ à 1 ruban

$$\mathcal{U}(\langle M \rangle \# x) \equiv M(x)$$

# la machine universelle

$\mathcal{U}$ a 3 rubans

1. entrée $\langle M \rangle \# x$
2. état simulé de $M$ en unaire
3. ruban de travail de $M$

. copier $x$ sur $3.$

$q_0 = 0$ $\quad 2.$

. simuler $M$ en lisant $2.$ , $3.$ puis

en cherchant dans $\langle M \rangle$ sur $1.$

une transition applicable

# la thèse de Church - Turing

algorithmes $\equiv$ MT $\equiv$ programmes

MT

binaire

plusieurs zones mémoires

code source

interprétation

opérations classiques ( + , × , ... )

# Langages récursivement énumérables (RE)
## semi - décidables

$\mathcal{L}(M) = \{ x \in \Sigma^* \mid M(x) \text{ accepte} \}$

ie $q_0 x \vdash_M^* y\, q_a\, z$

$L$ RE ssi $\exists$ MT $M$, $L = \mathcal{L}(M)$

on dit que $M$ accepte $L$

notons que si $x \notin L$, $M(x)$ refuse ou boucle

# fermeture

RE est stable par $\cup$, $\cap$

soient $L_1$, $L_2$ accepté par $M_1$, $M_2$ ; $M$ pour $L_1 \cup L_2$

idée : considérer $M$ à deux rubans

rubans $i$ pour $M_i$

sur une entrée $x$

- copier $x$ sur le ruban 1
- calculer $M_1(x)$ sur le ruban 1
  si acceptant, $M$ accepte — mais si ça plante ?
- puis calculer $M_2(x)$ sur le ruban 2
  si acceptant, $M$ accepte
- sinon, rejetter

solution : l'entrelacement

étape 1 de $M_1$ → étape 1 de $M_2$ →
étape 2 de $M_1$ → étape 2 de $M_2$ ...

plus de plantage

$\cap$ : idem ; plus simple car pas besoin d'entrelacement

# pourquoi "récursivement énumérable"

$M$ énumère $L$ ssi $M$ écrit en sortie (sans jamais rien effacer) une suite $m_0 \# m_1 \# \dots$ (possiblement infinie) telle que :

$\forall i, \ m_i \in L$

$\forall x \in L, \ \exists! \, i, \ m_i = x$

L est RE ssi $\exists$ M' énumérant L

$\mathcal{L}(M) = L$ rq : $M \neq M'$ possiblement

=> idée : considérer M' à 3 rubans

sur 1) incrémenter les mots binaires

2) simuler M sur chaque mot

et si ça plante ?

3) si le mot est accepté l'écrire ici

une infinité de calculs peut-être infinis

comment voir chaque étape de chaque calcul ?

on utilise un entrelacement diagonal

| | | étapes | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| mots | 1 | 1 | 3 | 6 |
| | 2 | 2 | 5 | |
| | 3 | 4 | | |

ordre d'exécution des opérations

$\Leftarrow$ si $M'$ énumère $L$

considérer $M$ à 2 rubans

1) entrée $x$

2) simule $M'$

si on voit passer $m_i = x$ sur le ruban 2)

accepter

un langage non . RE

$\mathcal{D} = \{ \langle M \rangle \mid M(\langle M \rangle)$ n'accepte pas $\}$

supposons que $D$ accepte $\mathcal{D}$

si $D(\langle D \rangle)$ n'accepte pas

$D \in \mathcal{D}$ donc $D(\langle D \rangle)$ accepte absurde

si $D(\langle D \rangle)$ accepte même contradiction

# Langages récursifs (R)
# décidables

L est récursif ssi $\exists$ M tq $\mathcal{L}(M) = L$
et M s'arrête tjrs

ie $x \in L$ ssi $M(x)$ accepte
$\notin$ refuse

notons que R => RE

# fermeture

$R$ est fermé par $\cup$ et $\cap$ ( idem RE )

mais aussi par $^c$ ( pas comme RE )

il suffit d'inverser la réponse

## une propriété

si $L$ est RE et ${}^cL$ aussi

$\Rightarrow$ $L$ est R

soient $M_1$ acceptant $L$

$M_2$ acceptant ${}^cL$

soit $M$ simulant $M_1$ et $M_2$ entrelacée

si $M_1(x)$ accepte $\quad$ $M$ accepte

$M_2(x)$ $\quad$ $M$ rejette

} tjrs ds un de ces cas

un langage non R

$\mathcal{H} = \{ \langle M \rangle \# x \mid M(x)$ s'arrête $\}$
est RE mais pas R

supposons que H accepte $\mathcal{H}$

soit H' tq H'(y) accepte ssi H ( y # y ) rejette
plante ssi H ( y # y ) accepte

si H'(<H'>) accepte alors H(<H'> # <H'>)
rejette donc H'(<H'>) ne s'arrête pas

absurde !

idem si $H'(\langle H' \rangle)$ plante