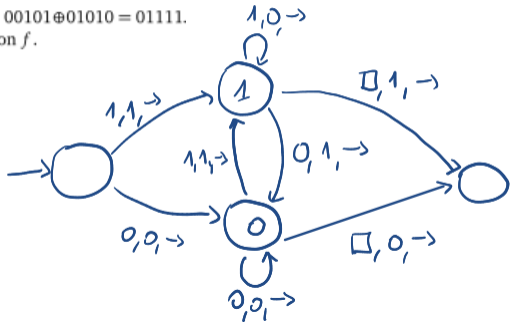


1. Soit $f : \{0, 1\}^* \rightarrow \{0, 1\}^*$ définie par $f(w) = (0w) \oplus (w0)$ où $0w$ et $w0$ sont les mots obtenus en ajoutant un 0 à gauche ou à droite de w respectivement, et \oplus est le XOR bit-à-bit.

Exemples : $f(110) = 0110 \oplus 1100 = 1010$, $f(0101) = 00101 \oplus 01010 = 01111$.

Décrire une machine de Turing qui calcule la fonction f .

$$\begin{array}{r} \\ \oplus \\ \hline 1 \end{array}$$



2. Étant donné un entier pair n , on souhaite calculer $n/2$ avec l'idée suivante : $n/2$ est le nombre de fois qu'il faut retrancher 2 à n pour obtenir 0.

- Écrire un programme WHILE¹ qui étant donné n pair², calcule $n/2$.
- Traduire ce programme WHILE en machine RAM.

i. \boxed{E} $x_1 = n$ (pair)

\boxed{S} $x_0 = n/2$

0. $x_0 \leftarrow 0$

1. WHILE $x_1 \neq 0$:

2. $x_1 \leftarrow x_1 - 1$

3. $x_1 \leftarrow x_1 - 1$

4. $x_0 \leftarrow x_0 + 1$

$n=4$

x_1	x_0
4	0
2	1
0	2

ii. \boxed{E} $R_1 = n$ (pair)

\boxed{S} $R_0 = n/2$

1. JUMP($R_1, 6$)

2. DEC(R_1)

3. DEC(R_1)

4. INC(R_0)

5. JUMP($R_2, 1$) \leftarrow JUMP(1) (inconditionnel)

6. STOP

Exercice 2 (sur 4 pts).

Vrai ou faux

Déterminer si chaque affirmation suivante est vraie ou fausse, en **justifiant la réponse**. Tout résultat du cours peut être utilisé en l'énonçant clairement, mais une réponse vrai/faux sans justification ne donne pas de point.

1. Il y a plus de fonctions calculables par des machines de Turing à 2 rubans que par des machines RAM.
2. La fonction f définie nulle part, c'est-à-dire telle que $f(w)$ est indéfini pour tout w , est une fonction calculable.
3. Une machine de Turing *cyclique* est une variante de machine de Turing dont le ruban est un *cycle* : ses cases sont numérotées de 0 à $n-1$; pour $0 \leq i < n-1$, la case $i+1$ est à droite de la case i , et la case 0 est à droite de la case $n-1$. Une machine de Turing cyclique peut calculer toute fonction calculable.
4. Il existe des langages décidables qui ne sont pas co-reconnaissables.

1. FAUX : les machines à 2 rubans sont équivalentes aux machines standard et les machines RAM sont équivalentes aux machines de Turing.

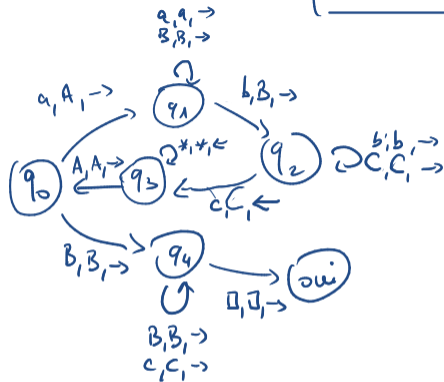
2. VRAI : Algorithme : $\text{def } f(x):$
 while true: $\text{if } x: f(x)$
 pass

3. FAUX : Mémoire bornée

4. FAUX : Décidable = reconnaissable + co-reconnaissable

q_0	a	A	\rightarrow	q_1
	B	B	\rightarrow	q_4
q_1	a	a	\rightarrow	q_1
	B	B	\rightarrow	q_1
	b	B	\rightarrow	q_2
q_2	b	b	\rightarrow	q_2
	C	C	\rightarrow	q_2
	c	C	\rightarrow	q_3
q_3	*	*	\rightarrow	q_3
	A	A	\rightarrow	q_0
q_4	B	B	\rightarrow	q_4
	C	C	\rightarrow	q_4
	\square	\square	\rightarrow	OUI

$\ast = a, b, c, B, C$



1. Déterminer si chacune des entrées suivantes est acceptée par \mathcal{M} : a, ac, abc, aabc, abca, aabbcc.

a: $q_0: \checkmark \vdash q_1: A \square \downarrow$ X

ac: $q_0: \checkmark \vdash q_1: A \checkmark \downarrow$ X

abc: $q_0: \checkmark \vdash q_1: A \checkmark \vdash q_2: AB \checkmark \vdash q_3: A \checkmark B C$
 $\vdash^* q_0: A \checkmark B C \vdash q_4: AB \checkmark C \vdash^* \text{oui} : \dots$ ✓

aabc: $q_0: \checkmark \vdash aabc \vdash^* q_0: A \checkmark \vdash BC \vdash q_1: AA \checkmark \vdash C \vdash q_1: AA \checkmark B \checkmark C \downarrow$ X

abca: $q_0: \checkmark \vdash abca \vdash^* q_0: A \checkmark \vdash Bca \vdash q_4: AB \checkmark \vdash ca \vdash q_4: ABC \checkmark \downarrow$ X

aabbcc: $q_0: \checkmark \vdash aabbcc \vdash^* q_0: A \checkmark \vdash BCC$

$\vdash^* q_0: AA \checkmark \vdash BCC \vdash q_4: AA \checkmark B \checkmark C C$

$\vdash^* q_4: AA \checkmark B \checkmark C C \checkmark \downarrow \vdash \text{oui} : AA \checkmark B \checkmark C C \checkmark \downarrow$ ✓

On note $w \in \{a, b, c\}^*$ le mot en entrée, et on l'écrit sous la forme $w = a^l b^m c^n w'$ avec $l, m, n \geq 0$. Exemples. $aabcbac = a^2 b^1 c^2 w'$ avec $w' = bac$, $bbcb = a^0 b^2 c^1 w'$ avec $w' = b$, $aac = a^2 b^0 c^1 w'$ avec $w' = \varepsilon$ (le mot vide).

2.
 - i. Montrer que si l, m ou n vaut 0, alors la machine se bloque dans l'un des états q_0, q_1 ou q_2 .
 - ii. Si $l > 0, m > 0$ et $n > 0$, la machine revient dans l'état q_0 : décrire la configuration lors du (premier) retour en q_0 .
3. On suppose maintenant $l, m, n > 0$.
 - i. Montrer que si $m < l$ ou $n < l$, alors la machine se bloque dans l'un des états q_1, q_2 ou q_4 .
 - ii. Montrer que si $m \geq l$ et $n \geq l$, alors la machine atteint l'état q_4 , et décrire la configuration obtenue.
4. Décrire aussi précisément que possible le langage accepté par M .

2. i. $l=0$: bloqué en q_0
 $m=0$: bloqué en q_1
 $n=0$: — q_2

ii. On remplace le 1^{er} a par A en passant de q_0 à q_1
 — b par B — q_1 à q_2
 — c par C — q_2 à q_3

Donc la configuration est : $q_0: A^l a^{l-1} B^m b^{m-1} C^n c^{n-1} w'$

4. On atteint "oui" si on ne lit que des B et C. Donc on $m=l, n=l, w' = \varepsilon$. Donc $L = \{a^l b^l c^l : l > 0\}$

3. i. Après m tours (si $m < l$ et $n \geq m$), on est dans la configuration $q_0: A^m a^{l-m} B^m C^m c^{n-m} w'$

On passe en $q_1: A^{m+1} a^{l-m-1} B^m C^m c^{n-m} w'$

et on est bloqué

• Si $m \leq n < l$: on atteint la configuration

$q_0: A^n a^{l-n} B^n b^{m-n} C^n w'$

$\hookrightarrow q_2: A^{n+1} a^{l-n-1} B^{n+1} b^{m-n-1} C^n w'$

• $l = m > n$: \rightarrow blocage en q_4

ii. Une fois remplacés tous les a en A, on atteint $q_0: A^l B^l b^{m-l} C^n c^{n-l} w'$

On passe en $q_4: A^l B^l b^{l-1} b^{m-l} C^n c^{n-l} w'$