NUMÉRO D'ANONYMAT : ....

MCAL MT – Examen

Durée : 1h30, sans document

- N'oubliez pas d'indiquer votre numéro d'anonymat sur le sujet puis glissez le dans votre copie à la fin de l'épreuve.
- Répondez sur votre copie sauf pour les questions avec pointillés.
- Commencez par lire tout le sujet pour repérer les questions faciles.
- Respectez les notations du cours.
- Le sujet est sur $(10100)_2$ points et comporte $(101)_2$ exercices indépendants.
- Le barème est donné à titre indicatif.
- Tous les appareils électroniques sont interdits à l'exception des montres qui ne communiquent pas.

Exercice 1 : Résultats du cours (5min, 2 pt)

Rappelez en quelques lignes un résutlat important du cours pour chacun des thèmes ci-dessous

1. CALCULABILITÉ

SOLUTION

- *Les fonctions calculables (par un algorithme en temps fini) sont les machines de Turing qui terminent pour toute entrée binaire.*
- *La notiont de calculabilité est indépendante du langage de programmation : tous les modèles de calculs existants ont la même puissance puisqu'on a trouvé à chaque fois une traduction des uns dans les autres.*

2. ENSEMBLE DÉNOMBRABLE / ENSEMBLE NON-DÉNOMBRABLE

SOLUTION

- *Les ensembles $\mathbb{N}^i$ pour $i \in \mathbb{N}$ sont dénombrables et l'ensemble des listes d'entiers $\bigcup_{i\in\mathbb{N}} \mathbb{N}^i$ est dénombrable.*
- *Les ensembles $\mathbb{N} \to \mathbb{N}$, $\mathbb{N} \to Bool$, $\mathscr{P}(\mathbb{N})$ ne sont pas dénombrables.*

Exercice 2 : Preuve d'indécidabilité associée à une réduction (5min, 2.5 pt)

Considérons un langage L sur l'alphabet Σ et un langage L' sur l'alphabet Σ' accompagnés d'un diagramme de réduction de L à L' qui satisfait l'équivalence (‡) :

$$\begin{array}{ccc} \omega \in \Sigma^* & \xrightarrow[traduction]{M_R} & R(\omega) = \omega' \in \Sigma'^* \\ \omega \in L & \underset{(\ddagger)}{\Longleftrightarrow} & R(\omega) \in L' \end{array}$$

On souhaite démontrer que « L non-reconnaissable $\Longrightarrow$ L' non-reconnaissable. »

Q1. Complétez la preuve (2 pt) Pour démontrer l'implication on suppo L non-reconnaissable et on doit montrer que L' e

Preuve par contradiction: Faisons l'hy L' e MT M' » et utilisons M' pour construire une MT M qui reconnaît L. On aura alors une contradiction puisque L e

L'hy M' reconnaît L' » signifie $M'(\omega') = \mathbb{V} \underset{(1)}{\iff} \omega' \in L'$.

Prenons $M \stackrel{def}{=} [M_R\ ;\ M']$. On doit montrer que M reconnaît L ce qui revient à montrer que $M(\omega) = \mathbb{V} \iff \omega \in L$. Pour cela il suffit de compléter le diagramme d'équivalence suivant :

$$\begin{array}{ccc}
M(\omega) = \mathbb{V} & \stackrel{?}{\iff} & \omega \in L \\
\Updownarrow \text{ def M} & & \Uparrow \\
\overbrace{\omega \xrightarrow{M_R} \underbrace{R(\omega) \xrightarrow{M'} \circledcirc}_{M'(R(\omega))=\mathbb{V}}} & & \text{d'après } (\ddagger) \\
 & & \Downarrow \\
 & \underset{(1)}{\iff} & R(\omega) \in L'
\end{array}$$

Conclusion : en suppo L' reconnaissable on aboutit à L reconnaissable ce qui contredit l'hy L non-reconnaissable. On obtient donc une contradiction ce qui prouve que L' e □

Exercice 3 : Applications du Théorème de Rice (25min, 6 pt)

Q2. (0.5 pt) **Complétez :** Un mot ω appartient au langage d'une MT M si $M(\omega) = \mathbb{V}$ c'est-à-dire $M(\omega) \to \circledcirc$. À l'inverse, ω n'appartient pas au langage d'une MT M si
$M(\omega) \to \otimes$ *ou* $M(\omega) \to \infty$

Q3. (0.25 pt) **Complétez :** Deux MT M_1 et M_2 sont équivalentes si et seulement si elles reconnaissent le même langage .

Q4. (0.25 pt) **Complétez :** Un ensemble de machines de Turing est un ensemble de Rice s'il s'écrit $\{m \mid \mathcal{C}(\mathscr{L}(U(m)))\}$ où $\mathcal{C}$ est une condition portant sur le langage reconnu par m

Q5. (0.25 pt) **Complétez :** Soit $\mathcal{M}$ l'ensemble des codages binaires des machines de Turing. Tout ensemble de Rice non-trivial c'est-à-dire différent de $\mathcal{M}$ et de $\emptyset$, est indécidable.

Q6. (0.25 pt) Si un ensemble L n'est pas un ensemble de Rice, que peut-on en déduire ?

SOLUTION

Le théorème de Rice donne un critère pour détecter des ensembles indécidables. Si un ensemble ne satisfait pas ce critère, on ne peut rien dire : il peut être décidable ou indécidable.

Q7. (1.25 pt) **Formalisez en termes mathématiques les ensemble suivants**

1. L'ensemble des machines de Turing équivalentes à la MT M

   $L_1 = \{m \in \mathcal{M} \mid \mathscr{L}(U(m)) = \mathscr{L}(M)\}$

2. L'ensemble des machines de Turing qui ne terminent pas pour le mot 00

   $L_2 = \{m \in \mathcal{M} \mid \forall \omega \in \{0,1\}^*,\ U(m)(00) \to \infty\}$

3. L'ensemble des MT qui terminent dans l'état $\otimes$ sur le mot binaire 00

$L_3 = \{ m \in \mathcal{M} \mid U(m)(00) = \mathbb{F} \}$

4. l'ensemble des MT qui n'acceptent pas le mot binaire 00

   $L_4 = \{ m \in \mathcal{M} \mid U(m)(00) \neq \mathbb{V} \}$

5. L'ensemble des machines de Turing qui n'acceptent aucun mot

   $L_5 = \{ m \mid \mathscr{L}(U(m)) = \emptyset \}$

6. L'ensemble des machines de Turing qui reconnaissent un nombre pair de mots

   $L_6 = \{ m \mid \exists k \in \mathbb{N},\ |\mathscr{L}(U(m))| = 2k \}$

Parmi les ensembles L_1 à L_6, 4 sont des ensembles de Rice, 2 n'en sont pas.

Q8. (2 pt) Pour justifier les 4 ensembles de Rice, vous rédigerez votre réponse de la manière suivante en définissant la condition de Rice, $\mathcal{C}$, correspondant à l'ensemble.

Indication : … est un ensemble de Rice car il peut s'écrire $\{ m \in \mathcal{M} \mid \mathcal{C}(\ \mathscr{L}(U(m))\ ) \}$ avec $\mathcal{C}(L) \stackrel{def}{=} \ldots$

SOLUTION

— *L'ensemble L_1 des machines de Turing équivalentes à la MT M est un ensemble de Rice car*

$$L_1 = \{ m \in \mathcal{M} \mid \mathscr{L}(U(m)) = \mathscr{L}(M) \} = \{ m \in \mathcal{M} \mid \mathcal{C}(\mathscr{L}(U(m))) \} \text{ avec } \mathcal{C}(L) \stackrel{def}{=} L = \mathscr{L}(M)$$

— *L'ensemble L_4 des MT qui n'acceptent pas le mot binaire 00 est un ensemble de Rice car*

$$L_4 = \{ m \in \mathcal{M} \mid U(m)(00) \neq \mathbb{V} \} = \{ m \in \mathcal{M} \mid \mathcal{C}(\mathscr{L}(U(m))) \} \text{ avec } \mathcal{C}(L) \stackrel{def}{=} 00 \notin L$$

— *L'ensemble L_5 des machines de Turing qui n'acceptent aucun mot est un ensemble de Rice car*

$$L_5 = \{ m \mid \mathscr{L}(U(m)) = \emptyset \} = \{ m \in \mathcal{M} \mid \mathcal{C}(\mathscr{L}(U(m))) \} \text{ avec } \mathcal{C}(L) \stackrel{def}{=} L = \emptyset$$

— *L'ensemble des MT qui reconnaissent un nombre pair de mots est un ensemble de Rice car*

$$L_6 = \{ m \mid \exists k \in \mathbb{N},\ |\mathscr{L}(U(m))| = 2k \} = \{ m \in \mathcal{M} \mid \mathcal{C}(\mathscr{L}(U(m))) \} \text{ avec } \mathcal{C}(L) \stackrel{def}{=} \exists k \in \mathbb{N},\ |L| = 2k$$

Q9. (1 pt) Expliquez pourquoi les 2 autres ensembles ne sont pas des ensemble de Rice.

SOLUTION

L_2 et L_3 ne sont pas des ensembles de Rice. Pour le voir, commençons par rappeler que le mot 00 appartient au langage d'une MT M si $M(00) \to \circledcirc$. À l'inverse, $00 \notin \mathscr{L}(M) \Leftrightarrow M(00) \to \otimes$ ou $M(00) \to \infty$.

$$\begin{aligned} L_4 &= \{ m \in \mathcal{M} \mid 00 \notin \mathscr{L}(U(m)) \} \\ &= \{ m \in \mathcal{M} \mid U(m)(\epsilon) \to \otimes \vee U(m)(\epsilon) \to \infty \} \\ &= \underbrace{\{ m \in \mathcal{M} \mid U(m)(\epsilon) \to \otimes \}}_{L_3} \cup \underbrace{\{ m \in \mathcal{M} \mid U(m)(\epsilon) \to \infty \}}_{L_2} \end{aligned}$$

L_4 est un ensemble de Rice puisqu'il s'écrit $\{ m \in \mathcal{M} \mid \mathcal{C}(\mathscr{L}(U(m))) \}$ avec $\mathcal{C}(L) \stackrel{def}{=} 00 \notin L$ mais ce n'est pas le cas pour L_2 et de L_3 car leur condition ne se ramène pas à une condition sur le langage reconnu par m.

Exercice 4 : Puissance des modèles de calcul (30min, 5 pt)

Répondez aux questions suivantes par vrai/faux et **justifiez votre réponse. Une réponse vrai/faux sans justification ne donne pas de point.**

1. Il existe des ensembles qui sont reconnaissables par un programme C mais qui ne sont pas reconnaissables par une MT à une bande.

   SOLUTION

   FAUX *puisque tout ce qui est calculable par un algorithme qui termine l'est par une MT. On pourrait traduire tout programme C travaillant sur une mémoire adressée en une MT équivalente travaillant sur un ruban. Et on peut coder une MT en C. Donc les deux modèles C et MT sont équivalents.*

2. Soit $L = \{\omega_1, \ldots, \omega_n\}$ un langage fini, il existe une automate (à nombre) d'états fini qui le reconnaît.

SOLUTION

VRAI. *Il suffit de construire un automate (à nombre) d'états fini avec une branche* $1 \xrightarrow{\omega_i} \circledcirc$ *pour chaque mot de* L et 1 *pour état initial.*

3. Soit L un langage fini, il existe une machine de Turing qui le reconnaît.

SOLUTION

VRAI. *Il suffit de construire la* MT *qui simule l'*AEF *de la question précédente.*

4. L'ensemble $\{0, 1\}^*$ des mots binaires est reconnaissable par une machine de Turing.

SOLUTION

VRAI. $M \stackrel{def}{=} \rightarrow \circledcirc$ *opérant sur l'alphabet* $\Sigma = \{\Box, 0, 1\}$ *reconnaît* $\{0, 1\}^*$.

5. L'ensemble $\mathcal{M}$ des codages binaires de machines de Turing est reconnaissable par une MT.

SOLUTION

VRAI. *Le format des codages binaires des machines de Turing peut se décrire par une expression régulière (cf. cours). donc* $\mathcal{M}$ *est reconnaissable par un automate (à nombre) d'états fini et donc par une machine de Turing.*

6. Soit L un langage infini, il n'existe pas de machine de Turing qui le reconnaît.

SOLUTION

FAUX. *Contre-exemple : L'ensemble infini* $\{ a^n \mid n \in \mathbb{N} \}$ *correspond à l'expression régulière* a^* *il est reconnaissable par un automate (à nombre) d'états fini et un automate (à nombre) d'états fini peut se coder sous forme de machine de Turing donc il existe une machine de Turing qui reconnaît un ensemble infini.*

7. Soit L un langage infini, il existe forcément une machine de Turing qui le reconnaît.

SOLUTION

FAUX. *Contre-exemple : Les langages* $\overline{L_{EF}} = \{ (m, \omega) \mid U(m)(\omega) \to \infty \}$ *n'est pas reconnaissable par une* MT *(cf. cours) et ce langage contient une infinité d'éléments. En effet, prenons une* MT $M_\infty \stackrel{def}{=} \rightarrow 1 \xrightarrow{\Sigma:R} 1$ *qui ne termine jamais. L'ensemble* $\{ (m_\infty, \omega) \mid m_\infty = [M_\infty]_2,\ \omega \in \{0, 1\}^* \}$, *qui est infini puisqu'il existe une infinité de mot binaire, est inclus dans* $\overline{L_{EF}}$.

8. Si une machine de Turing reconnaît un langage, alors il existe forcément une machine de Turing qui reconnaît son complémentaire.

SOLUTION

FAUX. *Contre-exemple : La machine de Turing universelle* U *reconnaît le langage universel*

$$L_U = \{ (m, \omega) \mid U(m, \omega) = \mathbb{V} \} = \mathscr{L}(U).$$

Or $\overline{L_u}$ *n'est pas reconnaissable par une* MT *(cf. cours).*

9. Il existe des algorithmes qu'on peut programmer avec une machine de Turing à deux bandes mais pas avec une machine de Turing à une seule bande.

SOLUTION

FAUX *puisqu'on peut traduire une* MT *à deux bandes en une* MT *à une seule bande (cf. TD2) et une machine de Turing à une bande peut-être vue comme une* MT *à deux bandes (dont on n'utilise pas le seconde bande).*

10. Les machines de Turing à plusieurs bandes reconnaissent plus de langages que les machines de Turing à une bande.

SOLUTION

FAUX *puisqu'on a justifié dans une question précédente qu'il y avait équivalence entre les* MT *à une ou deux bandes.*

Exercice 5 : Algorithme de substitution réalisé par une MT (25min, 4.5 pt)

On considère l'alphabet $\Sigma = \{\square, a, b, A\}$ **sans le symbole** \$ de début de bande. **La partie utile d'une bande** est la partie située entre l'infinité de $\square$ à gauche et l'infinité de $\square$ à droite. Au départ la tête de lecture/écriture est placée sur le premier symbole du mot. Une bande contenant le mot *abaaaba* sera donc de la forme $\begin{array}{c|c|c|c|c|c|c|c|c} {}^{\infty}\square & a & b & a & a & a & b & a & \square^{\infty} \\ & \uparrow & & & & & & & \end{array}$.

Q10. (0.75 pt) Donnez une MT $M_{\overleftarrow{début}}$ à une bande qui, à partir d'une position dans la partie utile de la bande, déplace la tête de lecture/écriture sur le début du mot c'est-à-dire sur le symbole non-$\square$ le plus à gauche de la bande. Elle termine dans un état $\circledcirc$ s'il existe et dans un état $\otimes$ si la bande est vide.

SOLUTION

$$M_{\overleftarrow{début}} =\!\!\rightarrow \begin{cases} 1 \xrightarrow{\square:H} \otimes \\ 1 \xrightarrow{\Sigma\setminus\{\square\}:H} 2 \xrightarrow{\Sigma\setminus\{\square\}:L} 2 \xrightarrow{\square:R} \circledcirc \end{cases}$$

Q11. (0.75 pt) Donnez une MT $M_{prefix}(B_1, B_2)$ à deux bandes qui décide si le mot de B_2 est un préfixe du mot de B_1 (autrement dit le début du mot de B_1 correspond au mot de B_2). On suppose qu'au départ les têtes de lecture/écriture de B_1 et B_2 pointent sur le début de leur mot.

Exemples :

- La MT M_{prefix} doit répondre $\mathbb{V}$ dans le cas $\begin{array}{r|c|c|c|c} B_1 = & {}^{\infty}\square & \omega & \ldots & \square^{\infty} \\ \hline B_2 = & {}^{\infty}\square & \omega & \square & \square^{\infty} \end{array}$
- La MT M_{prefix} doit répondre $\mathbb{F}$ dans le cas $\begin{array}{r|c|c|c|c} B_1 = & {}^{\infty}\square & \omega' & \ldots & \square^{\infty} \\ \hline B_2 = & {}^{\infty}\square & \omega & \square & \square^{\infty} \end{array}$ si les mots ω et ω' ne sont pas identiques
- La MT M_{prefix} doit répondre $\mathbb{F}$ dans le cas $\begin{array}{r|c|c|c|c} B_1 = & {}^{\infty}\square & \omega & \square\ldots\square & \square^{\infty} \\ \hline B_2 = & {}^{\infty}\square & \omega & \omega' & \square^{\infty} \end{array}$

SOLUTION

$$M_{prefix} =\!\!\rightarrow \begin{cases} 1 \xrightarrow[\ell\in\Sigma\setminus\{\square\}]{(1,2)\ \ell:R} 1 \\ 1 \xrightarrow[(2)\ \square:H]{(1)\ \Sigma:H} \circledcirc \\ 1 \xrightarrow[\ell_1\in\Sigma,\ell_2\in\Sigma\setminus\{\square\},\ell_1\neq\ell_2]{(1)\ell_1:H\ \ (2)\ell_2:H} \otimes \end{cases}$$

Q12. (0.75 pt) Donnez une MT $M_{\overrightarrow{copy}}(B_2, B_1)$ a deux bandes qui copie le contenu de B_2 sur B_1. On suppose qu'au départ les têtes de lecture/écriture de B_1 et B_2 pointent sur le début de leur mot.

Exemple :

$$\begin{array}{r|c|c|c|c|c} & & \downarrow & & & \\ B_1 = & {}^{\infty}\square & a & a & a & \square^{\infty} \\ \hline B_2 = & {}^{\infty}\square & b & b & \square & \square^{\infty} \\ & & \uparrow & & & \end{array} \xrightarrow{M_{\overrightarrow{copy}}} \begin{array}{r|c|c|c|c|c} B_1 = & {}^{\infty}\square & b & b & \square & \square^{\infty} \\ \hline B_2 = & {}^{\infty}\square & b & b & \square & \square^{\infty} \end{array}$$

SOLUTION

$$M_{\overrightarrow{copy}} =\!\!\rightarrow \begin{cases} 1 \xrightarrow{(1,2)\square:H} \circledcirc \\ 1 \xrightarrow[\ell_1,\ell_2\in\Sigma,\ \ell_1\neq\square\vee\ell_2\neq\square]{(1)\ell_1/\ell_2:R\ \ (2)\ell_2:R} 1 \end{cases}$$

Algorithme de substitution À l'aide de MT précédentes on souhaite concevoir une machine de Turing $M_{subst}(B_1, B_2, B_3, B_4)$ à 4 bandes qui remplace les occurences du mot de B_2 dans le mot B_1 par le mot de B_3. Au départ la bande B_4 est vierge, elle sert d'espace de travail et les têtes de lecture/écriture de chaque bande pointent sur le début de leur mot.

Exemples :

- $M_{subst}(abaaaba, aa, A, \epsilon) = (abAaba, aa, A, \epsilon)$
- $M_{subst}(abaaaba, aa, a, \epsilon) = (abaaba, aa, a, \epsilon)$

Pour illustrer l'algorithme vous prendrez l'exemple suivant

		$\downarrow$						
$B_1 =$	${}^\infty\square$	a	b	a	a	b	a	$\square^\infty$
$B_2 =$	${}^\infty\square$	a	a	$\square$	$\square$	$\square$	$\square$	$\square^\infty$
$B_3 =$	${}^\infty\square$	A	$\square$	$\square$	$\square$	$\square$	$\square$	$\square^\infty$
$B_4 =$	${}^\infty\square$	$\square$	$\square$	$\square$	$\square$	$\square$	$\square$	$\square^\infty$
		$\uparrow$						

Q13. (2.25 pt) **Pour chaque étape de l'algorithme :**

— vous décrirez soigneusement, en français, ce qu'elle fait ;

— vous donnerez son effet sur les bandes B_1, B_2, B_3, B_4 de l'exemple ;

— vous donnerez le graphe des transitions de la MT correspondant à cette étape.

Indication : Respectez les conventions suivantes :

— Prenez soin de bien numéroter vos états 1, 2, 2$\mathbb{V}$, 2$\mathbb{F}$, 3, ... afin qu'on puisse recontruire le graphe en connectant les transistions.

— Une transition $\xrightarrow{(i)\ \ell/e:d}$ indique que l'action porte sur la bande B_i

— Une transition $\xrightarrow{M(B_i,B_j)}$ indique qu'on exécute la MT M sur les bandes B_i et B_j.

SOLUTION

1 *si on a atteint la fin du mot de B_1, on passe à l'état 3 sinon on compare B_1 et B_2 grâce à la MT M_{prefix} :*

$$1 \xrightarrow{(1)\ \square:H} 3$$

$$1 \xrightarrow{(1)\ \Sigma\backslash\square:H} 2 \xrightarrow{M_{prefix}(B_1,B_2)} \left\{\begin{array}{l} 2\mathbb{V} \\ 2\mathbb{F} \end{array}\right.$$

2 — *si $M_{prefix}(B_1, B_2) = \mathbb{F}$: on recopie* **le premier symbole** *de B_1 sur B_4 avant de reprendre la comparaison au symbole suivant. Pour cela, on replace T_1 au début de B_1, on recopie sur B_4 le premier symbole de B_1 qu'on remplace par $\square$ pour indiquer qu'on l'a traité ; on reprend à l'état 1 pour continuer la comparaison :*

$$2\mathbb{F} \xrightarrow{M_{\overleftarrow{début}}(B_1)} \circ \xrightarrow{M_{\overleftarrow{début}}(B_2)} \circ \xrightarrow[(4)\ \Sigma/\ell:R]{(1)\ \ell/\square:R} 1$$

On obtient :

			$\downarrow$					
$B_1 =$	${}^\infty\square$	$\square$	b	a	a	b	a	$\square^\infty$
$B_2 =$	${}^\infty\square$	a	a	$\square$	$\square$	$\square$	$\square$	$\square^\infty$
$B_3 =$	${}^\infty\square$	A	$\square$	$\square$	$\square$	$\square$	$\square$	$\square^\infty$
$B_4 =$	${}^\infty\square$	a	$\square$	$\square$	$\square$	$\square$	$\square$	$\square^\infty$
			$\uparrow$					

— *si $M_{prefix}(B_1, B_2) = \mathbb{V}$: on se trouve dans une configuration :*

						$\downarrow$		
$B_1 =$	${}^\infty\square$	$\square$	$\square$	a	a	b	a	$\square^\infty$
$B_2 =$	${}^\infty\square$	a	a	$\square$	$\square$	$\square$	$\square$	$\square^\infty$
$B_3 =$	${}^\infty\square$	A	$\square$	$\square$	$\square$	$\square$	$\square$	$\square^\infty$
$B_4 =$	${}^\infty\square$	a	b	$\square$	$\square$	$\square$	$\square$	$\square^\infty$
				$\uparrow$				

On recopie B_3 sur B_4 ; on replace T_3 au début de B_3 ; on replace T_2 au début de B_2 ; on efface ce qui précède la position courante sur B_1 et on se replace sur le premier symbole à traiter à l'aide de la MT $M_{forget}(B_1, B_2)$; on reprend à l'état 1 pour continuer la comparaison :

$$\textcircled{2\mathbb{V}} \xrightarrow{M_{\overleftarrow{début}}(B_1)} \bigcirc \xrightarrow{M_{\overleftarrow{début}}(B_2)} \bigcirc \xrightarrow{M_{\overrightarrow{copy}}(B_3,B_4)} \bigcirc \xrightarrow{M_{\overleftarrow{début}}(B_3)} \bigcirc \xrightarrow{M_{\overrightarrow{forget}}(B_1,B_2)} \bigcirc \xrightarrow{M_{\overleftarrow{début}}(B_2)} \textcircled{1}$$

où

$$M_{forget}(B_1, B_2) =\!\rightarrow \textcircled{1} \xrightarrow{(1)\Sigma/\Box:R \quad (2)\Sigma\backslash\{\Box\}:R} \textcircled{1} \xrightarrow{(2)\Box:H} \circledcirc$$

On obtient :

						$\downarrow$		
$B_1 =$	$^\infty\Box$	$\Box$	$\Box$	$\Box$	$\Box$	b	a	$\Box^\infty$
$B_2 =$	$^\infty\Box$	a	a	$\Box$	$\Box$	$\Box$	$\Box$	$\Box^\infty$
$B_3 =$	$^\infty\Box$	A	$\Box$	$\Box$	$\Box$	$\Box$	$\Box$	$\Box^\infty$
$B_4 =$	$^\infty\Box$	a	b	A	$\Box$	$\Box$	$\Box$	$\Box^\infty$
					$\uparrow$			

③ *À cette étape on se trouve dans la configuration suivante :*

							$\downarrow$	
$B_1 =$	$^\infty\Box$	$\Box$	$\Box$	$\Box$	$\Box$	$\Box$	$\Box$	$\Box^\infty$
$B_2 =$	$^\infty\Box$	a	a	$\Box$	$\Box$	$\Box$	$\Box$	$\Box^\infty$
$B_3 =$	$^\infty\Box$	A	$\Box$	$\Box$	$\Box$	$\Box$	$\Box$	$\Box^\infty$
$B_4 =$	$^\infty\Box$	a	b	A	b	a	$\Box$	$\Box^\infty$
							$\uparrow$	

On recopie B_4 sur B_1 puis on efface B_4 : $\textcircled{3} \xrightarrow{M_{\overleftarrow{début}}(B_4)} \bigcirc \xrightarrow{M_{\overrightarrow{copy}}(B_4,B_1)} \bigcirc \xrightarrow{M_{\overleftarrow{eff}}(B_4)} \bigcirc \xrightarrow{M_{\overleftarrow{début}}(B_1)} \circledcirc$

$$\text{avec } M_{\overleftarrow{eff}} =\!\rightarrow \left\{ \begin{array}{l} \bigcirc \xrightarrow{\Sigma\backslash\{\Box\}/\Box:L} \bigcirc \\ \bigcirc \xrightarrow{\Box:H} \circledcirc \end{array} \right.$$

							$\downarrow$					
$B_1 =$	$^\infty\Box$	$\Box$	$\Box$	$\Box$	$\Box$	$\Box$	a	b	A	b	a	$\Box^\infty$
$B_2 =$	$^\infty\Box$	a	a	$\Box$	$\Box$	$\Box$	$\Box$	$\Box$	$\Box$	$\Box$	$\Box$	$\Box^\infty$
$B_3 =$	$^\infty\Box$	A	$\Box$	$\Box$	$\Box$	$\Box$	$\Box$	$\Box$	$\Box$	$\Box$	$\Box$	$\Box^\infty$
$B_4 =$	$^\infty\Box$	$\Box$	$\Box$	$\Box$	$\Box$	$\Box$	$\Box$	$\Box$	$\Box$	$\Box$	$\Box$	$\Box^\infty$
		$\uparrow$										