

# Université de Paris 7

## Calculabilité et complexité— M1

### TD du 21 Septembre 2010 : Machines de Turing

**Exercice 1 — *Notion de problèmes*** On rappelle qu'un problème s'identifie à un langage, c'est-à-dire un ensemble de mots, sur une alphabet donné. Une telle identification définit un *codage* du problème.

Proposer des codages possibles de graphes finis qui à tout graphe associe un mot qui lui est propre. Dire comment on peut tester si un mot code bien un graphe et si oui comment ce graphe peut être déterminé sans ambiguïté.

**Exercice 2 — *Configurations*** Une configuration d'une machine de Turing est un couple formé du contenu de sa bande et de son état courant. On se propose de coder une telle configuration par un mot sur l'alphabet  $\Gamma \cup Q$  (on suppose sans perte de généralité que ces deux ensembles sont disjoints).

On décide par convention de placer l'état courant à gauche de symbole occupant la cellule sur laquelle est positionnée la tête de lecture. La configuration est alors codée par la plus petite suite de caractères sur le ruban, contenant tous les symboles distinct de B.

Donner l'ensemble des configurations possibles et montrer qu'il est reconnaissable par une automate fini que l'on décrira.

**Exercice 3 — *Fonction de transition*** Montrer que l'on n'augmente pas le pouvoir de calcul des machines de Turing si l'on suppose que la fonction de transition est seulement partiellement définie.

Pour les automates finis on ne peut pas supposer qu'il existe un unique état final (le prouver). Pour quelle raison, a priori, il n'en est pas de même pour les machines de Turing (on ne demande pas un raisonnement formel, seulement une intuition).

**Exercice 4 — *Calculer avec des machines de Turing*** 1) Construire une machine de Turing qui teste si un mot d'entrée sur l'alphabet  $\{0,1\}$  a deux fois plus d'occurrences de 0 que de 1. Tracer son diagramme et évaluer le nombre d'opérations que ce calcul comporte.

2) Construire une machine de Turing qui prend en entrée une suite de 0 et de 1 interprétée comme la représentation binaire d'un entier  $n > 0$  et qui s'arrête en laissant sur la bande la représentation binaire de  $n - 1$  (ex : à 10100 est associé 10011).

On pourra envisager la représentation habituelle avec le bit de poids fort à gauche et la représentation moins traditionnelle avec le bit de poids fort à droite.

On pourra aussi commencer par s'assurer que la représentation est standard (pas de bit de poids fort égal à 0).