IFT-2002

Informatique Théorique

H14 - cours 8

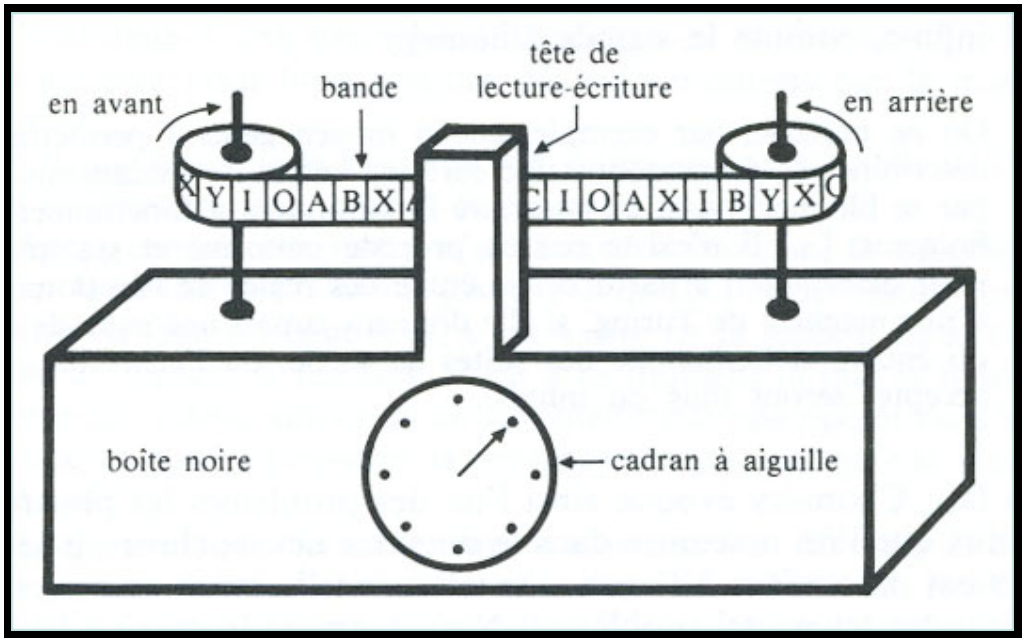
Julien Marcil - julien.marcil@ift.ulaval.ca

Aujourd'hui

- Machine de Turing
- La thèse de Church-Turing

Machine de Turing

Machine de Turing



Différences avec les automates finis

- La tête peut se déplacer dans les deux sens.
- C'est une tête de lecture/écriture.
- Le ruban est infini.
- Il y a deux états finaux, qui sont des états d'arrêt (aussitôt que la machine en atteint un, elle s'arrête).

À chaque étape

1. lit le symbole sur le ruban
2. fait une transition d'état
3. écri un symbole sur le ruban
4. déplace la tête vers la droite ou vers la gauche

Machine de Turing

Définition: Une **machine de Turing** consiste en un 7-tuple de la forme $(S, \Sigma, \Gamma, \delta, \iota, s_{\text{accepte}}, s_{\text{rejete}})$ où

- S est un *ensemble fini d'états*.
- Σ est *l'alphabet* d'entré.
- Γ est l'alphabet du ruban tel que $\_ \in \Gamma$ et $\Sigma \subseteq \Gamma$.
- $\delta : S \times \Gamma \to S \times \Gamma \times \{L, R\}$ est la *fonction de transition*.
- $\iota \in S$ est l'*état initial*.
- $s_{\text{accepte}} \in S$ est l'*état final acceptant*.
- $s_{\text{rejete}} \in S$ est l'*état final rejetant* et $s_{\text{accepte}} \neq s_{\text{rejete}}$

Remarque

- ␣ représente une case inoccupée du ruban: initialement, toutes les cases autres que celles qui contiennent la séquence d'entrée contiennent un ␣.

Example

Donner une machine de Turing M qui accepte le langage $\{0^{2^n} \mid n \geq 0\}$.

Algorithme

Sur l'entré w

1. Déplacer la tête vers la droite jusqu'à la fin du mot en remplacant un 0 sur deux par un X.
2. Si dans l'épape 1, le ruban ne contient qu'un seul 0 accepter w
3. Si dans l'épape 1, le ruban ne contient un nombre impaire de 0 (plus qu'un) alors rejetter w
4. Déplacer la tête vers la gauche jusqu'au début du mot.
5. retourner l'étape 1.

Example

Donner une machine de Turing M qui accepte le langage $\{w\#w \mid w \in \{0,1\}^*\}$.

Configuration

Définition: Une **configuration** d'une machine de Turing est donnée par

- son état
- le contenu de son ruban
- la position de sa tête de lecture

Notation

La description du ruban se fait en donnant la séquence de symboles du ruban en commençant par la première case.

Le symbole sous la tête de lecture/écriture est souligné.

Arrêt du machine de Turing

Une machine de turing peut:

- se rendre à un état final et arrêter
- boucler indéfiniment

Turing-acceptable

Définition: Un langage L est dit **Turing-acceptable** (ou *récursivement énumérable*)s'il existe une machine de Turing qui accepte les mots de L.

Étant donnée une entrée w

- Si $w \in L$ alors M s'arrête et accepte w
- Si $w \notin L$ alors M s'arrête et rejette w ou ne s'arrête pas

Turing-décidable

Définition: Un langage L est dit **Turing-décidable** (ou *décidable*)s'il existe une machine de Turing M qui accepte les mots de L et rejette les autres mots.

Étant donnée une entrée w

- Si $w \in L$ alors M s'arrête et accepte w
- Si $w \notin L$ alors M s'arrête et rejette w

Remarque

Par définition, tout les langages décidables sont également des langages Turing-acceptables.

Théorème

Soit un le langage L. Si L et $\overline{L}$ sont *Turing-acceptable*, alors L est *Turing-décidable*.

Variantes de machine de Turing

Il existe plusieurs variantes sur les machines de Turing:

- Machine de Turing à plusieurs rubans
- Machine de Turing à acceptation par message
- Machine de Turing non déterministe

Machine de Turing universelle

Une machine de Turing universelle est une machine de Turing qui peut simuler n'importe quelle machine de Turing sur n'importe quelle donnée d'entrée. La machine universelle réalise cela en lisant de son propre ruban la description de la machine à simuler et la donnée d'entrée de celle-ci.

Les plus petites machines de Turing universelles connues ont les tailles suivantes : 2×18, 3×10, 4×6, 5×5, 7×4, 10×3, 22×2.

Simulateurs

- http://morphett.info/turing/turing.html
- http://db.ing.puc.cl/turingmachine/

The LEGO Turing Machine

0:00 / 2:41

La thèse de Church-Turing

Qu’est-ce qu’un algorithme?

- Quelque chose qui peut être programmé. (Mais que veut dire “programmer”?)
- Un ensemble fini d’instructions permettant d’effectuer une tâche donnée.
- Une recette finie que l’on peut suivre pour obtenir en temps fini une réponse à une question donnée.

Quelques algorithmes

- Comment additionner/multiplier deux entiers avec plus que 1 chiffre.
- Algorithme d'Euclide (300 av. J.C) pour calculer le plus grand diviseur commun de deux entiers.
- Algorithme pour déterminer si une équation $ax^2 + bx + c = 0$ admet une solution.
- Historiquement, les algorithmes ont d'abord été des façons systématiques d'effectuer une série de calculs afin de résoudre une question mathématique.

Propriétés

Une notion formelle d'algorithme devrait répondre aux critères suivants:

- La description d'un algorithme est finie. On peut également désirer que cette description soit une liste d'instructions.
- Chaque étape d'exécution peut être effectuée en temps fini.
- Un algorithme reçoit une entrée finie et retourne une réponse de longueur finie.
- L'exécution d'un algorithme se termine toujours après un nombre fini d'étapes.

Propriétés

Une notion formelle d’algorithme devrait répondre aux critères suivants:

- L'algorithme peut en principe être suivi par un humain avec seulement du papier et un crayon
- L'exécution de l'algorithme ne requiert pas d'intelligence de l'humain sauf celle qui est nécessaire pour comprendre et exécuter les instructions.