Décidabilité - EISTI - ING 2

Yannick Le Nir

Ecole Internationale des Sciences du Traitement de l'Information

yannick.lenir@eisti.fr

Introduction

Complexité Informatique

- comment concevoir des algorithmes efficaces ?
- comment savoir qu'ils sont corrects et efficaces ?
- comment savoir que c'est peu probable qu'un algorithme efficace existe ?

Livre : Papadimitriou : Computational complexity
Livre : Cormen, Leiserson, Rivest et Stein : Introduction à l'algorithmique, Dunod

Algorithme et Complexité

Algorithme

Suite d'instructions permettant de passer des données initiales d'un problème au résultat final.

Complexité

Mesure de l'efficacité (performance) d'un algorithme

1. Nombre d'opérations élémentaires :
   1.1 opérations élémentaires sur les entiers et réels
   1.2 opérations sur vecteurs dans $\mathbb{R}^n$ (resp. sur matrices) définies à partir de n opérations élémentaires (resp. n^2)
2. Taille des données : nombre de bits nécessaires pour coder ces données dans la machine (ex : n sommets $\rightarrow n^2$ bits).

Complexité Formelle

Définition

Soit $\mathcal{A}$ un algorithme. Soit I une spécification des données (instance). Soit $f(\mathcal{A}, I)$ le nombre d'opérations élémentaires pour passer de I (taille n) au résultat de $\mathcal{A}$. Complexité de $\mathcal{A}$:

$$C_{\mathcal{A}}(n) = max\{f(\mathcal{A}, I) \mid |I| = n\}$$

Remarque

On se contentera en général d'un majorant asymptotique via la notation $\mathcal{O}(g(n))$ de Landau :

$$\exists K < +\infty \text{ et } N_k \in \mathbb{N} \text{ t.q. } \forall n \geq N_k,\ |C_{\mathcal{A}}(n)| \leq K.|(g(n))|$$

Introduction sur les Graphes

Motivations

- Informatique
- Recherche opérationnelle
- Etude des processus stochastiques

Introduction sur les Graphes

Motivations

- Informatique
- Recherche opérationnelle
- Etude des processus stochastiques

Reférences

- Claude Berge ("Graphes et HyperGraphes", Dunod 1970)
- Anne Dicky ("Cours au CNAM - Inform. 1999-2000)

Rappels

Rappels de complexité

Rappels sur les graphes

Rappels sur la logique

Rappels d'un machine de Turing

Représentation et fonctionnement d'une machine de Turing

Description formelle d'une machine de Turing

Définition d'un graphe

Graphe $G(S, A)$

Constitué de 2 ensembles distincts d'éléments :

- L'ensemble S des sommets : $S = \{s_1, s_2, \cdots, s_n\}$
- L'ensemble A des arcs : $A = \{a_1, a_2, \cdots, a_{N_A}\}$
  ensemble de paires non ordonnées de sommets distincts

Remarque

- Graphe orienté : orientation entre les 2 sommets extrémités de chaque arc $a_i \in A$
- Représentation graphique : points = sommets, segments ou flèches = arcs
- Deux sommets s_i et s_j sont adjacents si il existe un arc $a_l = \{s_i, s_j\}$

Rappels

Rappels de complexité

Rappels sur les graphes

Rappels sur la logique

Rappels d'un machine de Turing

Représentation et fonctionnement d'une machine de Turing

Description formelle d'une machine de Turing

Chemins dans les graphes

Rappels

Rappels de complexité

Rappels sur les graphes

Rappels sur la logique

Rappels d'un machine de Turing

Représentation et fonctionnement d'une machine de Turing

Description formelle d'une machine de Turing

Définitions

- Une séquence alternée de sommets et d'arcs d'un graphe G est appelée chemin
- Le nombre n d'arcs d'un chemin de G est appelé longueur de ce chemin
- Un chemin est fermé si le sommet extrémité initiale s_{in} est identique au sommet extrémité finale $s_{fin} \Leftrightarrow s_{in} = s_{fin}$
- Si tous les sommets d'un chemin sont différents on l'appelle chemin élémentaire

Théorème

Il existe un chemin entre deux sommets s_i et $s_j \Leftrightarrow$ Il existe un chemin élémentaire entre les sommets s_i et s_j

Chemins (suite)

Définitions

- Un chemin fermé tel que $\forall i \neq j,\ s_i \neq s_j$ (sauf $s_{in} = s_{fin}$) est appelé circuit. Si le circuit est de longueur k on l'appelle k-circuit
- Si un chemin passe une et une seule fois par tous les sommets d'un graphe G, on l'appelle chemin hamiltonien
- Si un chemin hamiltonien est un circuit, on l'appelle circuit hamiltonien

Correspondances entre graphes

Graphe orienté vs Graphe non orienté

arc	arête
chemin	chaîne
circuit	cycle
chemin élémentaire	chaîne élémentaire
chemin hamiltonien	chaîne eulérienne
circuit hamiltonien	cycle eulérien

Eléments du langage de la logique propositionnelle

Rappels
Rappels de complexité
Rappels sur les graphes
Rappels sur la logique
Rappels d'un machine de Turing
Représentation et fonctionnement d'une machine de Turing
Description formelle d'une machine de Turing

Langage formel $\mathcal{L}_0$

Les propositions seront représentées par des symboles de valeur de vérité vraie ou fausse :

- Ensemble V_p, au plus dénombrable, des propositions notées $p, q, \ldots$.
- Ensemble Ξ, au plus dénombrable, des constantes.
- Ensemble L des connecteurs :
  - logique unaires : la négation $\neg$
  - propositionnels binaires :
    - disjonction : $\vee$
    - conjonction : $\wedge$
    - implication : $\rightarrow$
    - équivalence : $\leftrightarrow$
- Séparateurs : parenthèses (,) et crochets [,].

Eléments du langage (suite)

Rappels
Rappels de complexité
Rappels sur les graphes
Rappels sur la logique
Rappels d'un machine de Turing
Représentation et fonctionnement d'une machine de Turing
Description formelle d'une machine de Turing

Définition

Un atome est un proposition dont la structure interne ne nous préoccupe pas. *Notation :* $p, q, r, \ldots$

Définition

Une formule bien formée (fbf) :

- un atome
- proposition obtenue à partir des fbf A et B :
  - $\neg A$
  - $A \vee B$, $A \wedge B$
  - $A \rightarrow B$, $A \leftrightarrow B$

Eléments du langage (suite)

Rappels
Rappels de complexité
Rappels sur les graphes
Rappels sur la logique
Rappels d'un machine de Turing
Représentation et fonctionnement d'une machine de Turing
Description formelle d'une machine de Turing

Notations

- $A_0 = \{V_p, \Xi, L\}$ (alphabet du langage)
- $F_0 = A, B, C, ...$ (ensemble des fbf)
- $\mathcal{L}_0 = \{A_0, F_O\}$ (langage d'ordre 0 : langage du calcul propositionnel)

Interprétation sémantique de la logique propositionnelle

Valeur de vérité

Donner un sens aux (fbf) : Tout atome peut prendre deux valeurs : vrai (1) et faux (0) et la valeur de vérité d'une fbf est complètement déterminée par la valeur de chacun de ses atomes.

Table de vérité de $\mathcal{L}_0$

p	q	$\neg p$	$p \vee q$	$p \wedge q$	$p \rightarrow q$	$p \leftrightarrow q$
0	0	1	0	0	1	1
0	1	1	1	0	1	0
1	0	0	1	0	0	0
1	1	0	1	1	1	1

Alphabets Langages

Rappels
Rappels de complexité
Rappels sur les graphes
Rappels sur la logique
Rappels d'un machine de Turing
Représentation et fonctionnement d'une machine de Turing
Description formelle d'une machine de Turing

Définitions

- Alphabet Σ : ensemble fini
- Σ^n : mots de longueur n
- $w = a_1 a_2 \cdots a_n \in \Sigma^n, \ |w| = n$: longueur du mot w
- ε : mot vide. $|\varepsilon| = 0$
- $\Sigma^* = \cup_{i \in \mathbb{N}} \Sigma^i$: les mots sur l'alphabet Σ
- Concaténation de $w, w' \in \Sigma^*$, notée ww'.
  $|ww'| = |w| + |w'|$

Historique

Machine de Turing

- Alan Matheson Turing : 1912-1954
- Rôle actif pour décrypter la machine Enigma pendant la seconde guerre mondiale
- Inventeur de la machine de Turing (1936)

Rappels

Rappels de complexité

Rappels sur les graphes

Rappels sur la logique

Rappels d'un machine de Turing

Représentation et fonctionnement d'une machine de Turing

Description formelle d'une machine de Turing

Machine de Turing

Généralités

- Tout algorithme peut être traduit en un programme pour la machine de Turing
- Modèle théorique de l'ordinateur (réalisations physiques dès 1940).

Description

- Ruban infini : suite de cases portant chacune un élément d'un alphabet
- Semblable aux automates finis, sauf qu'elle peut lire, écrire et se déplacer sur le ruban
- Déplacement d'une seule case (droite ou gauche) à la fois
- Diagramme de transition pour modéliser son comportement

Rappels
Rappels de complexité
Rappels sur les graphes
Rappels sur la logique
Rappels d'un machine de Turing
Représentation et fonctionnement d'une machine de Turing
Description formelle d'une machine de Turing

Machine de Turing

Représentation

1	1	0	1	1	0	1						$\cdots$

$\uparrow$

$\square$ Mécanisme de contrôle : nombre d'états fini

- Règles de transitions : (état initial,caractère lu,état final,caractère écrit,déplacement)
- $M(n)$: résultat en écriture

Rappels
Rappels de complexité
Rappels sur les graphes
Rappels sur la logique
Rappels d'un machine de Turing
Représentation et fonctionnement d'une machine de Turing
Description formelle d'une machine de Turing

Machine de Turing

Fonctionnement

Initialisation : Un mot est inscrit sur le ruban et la tête est positionnée sur le caractère le plus à gauche

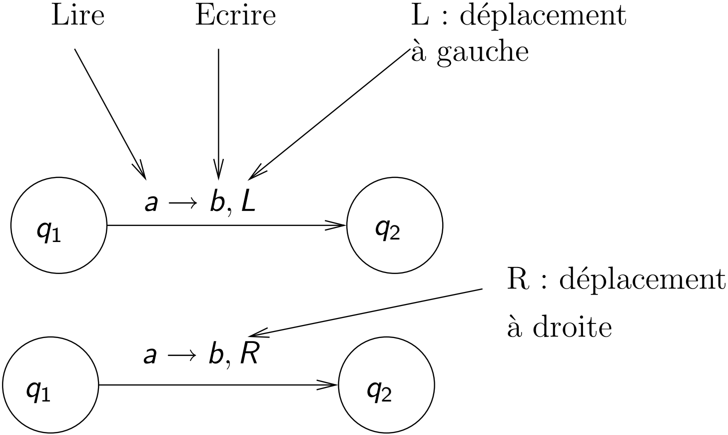
A chaque étape, la machine de Turing :

- lit un symbole
- fait une transition d'état
- fait l'une des trois actions suivantes :
  - écriture d'un symbole
  - déplacement de la tête vers la droite
  - déplacement de la tête vers la gauche

Rappels
Rappels de complexité
Rappels sur les graphes
Rappels sur la logique
Rappels d'un machine de Turing
Représentation et fonctionnement d'une machine de Turing
Description formelle d'une machine de Turing

Machine de Turing

Etats et Transitions



Rappels
Rappels de complexité
Rappels sur les graphes
Rappels sur la logique
Rappels d'un machine de Turing
Représentation et fonctionnement d'une machine de Turing
Description formelle d'une machine de Turing

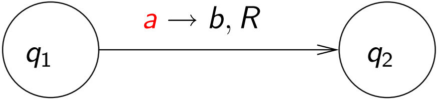
Machine de turing

Exemple de transition

a	b	a	c	$\diamond$	$\diamond$	$\diamond$						$\cdots$

$\uparrow$

q_1



Rappels

Rappels de complexité
Rappels sur les graphes
Rappels sur la logique
Rappels d'un machine de Turing
Représentation et fonctionnement d'une machine de Turing
Description formelle d'une machine de Turing

Machine de turing

Exemple de transition

a	b	b	c	$\diamond$	$\diamond$	$\diamond$						$\cdots$

$$\uparrow$$
$$q_2$$

$$q_1 \xrightarrow{a \to b, R} q_2$$

Rappels

Rappels de complexité

Rappels sur les graphes

Rappels sur la logique

Rappels d'un machine de Turing

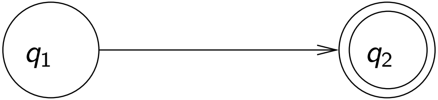
Représentation et fonctionnement d'une machine de Turing

Description formelle d'une machine de Turing

Machine de Turing

Acceptation et Langage

- La machine s'arrête s'il n'y a plus de transition possible
- Mot accepté si la machine s'arrête dans un état final :



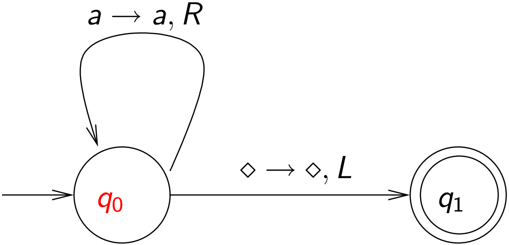
- Mot rejeté si la machine s'arrête dans un état non final ou entre dans une boucle
- L'ensemble des mots acceptés constituent le langage de la machine de Turing

Machine de Turing

Rappels
Rappels de complexité
Rappels sur les graphes
Rappels sur la logique
Rappels d'un machine de Turing
Représentation et fonctionnement d'une machine de Turing
Description formelle d'une machine de Turing

Exemple de Machine de Turing

Soit la machine suivante qui accepte le langage $L = a^*$:



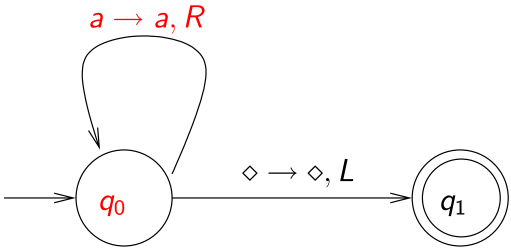
$T = 0$

Machine de Turing

Rappels
Rappels de complexité
Rappels sur les graphes
Rappels sur la logique
Rappels d'un machine de Turing
Représentation et fonctionnement d'une machine de Turing
Description formelle d'une machine de Turing

Exemple de Machine de Turing

Soit la machine suivante qui accepte le langage $L = a^*$:



a	a	a	$\diamond$	$\diamond$	$\diamond$	$\diamond$	$\cdots$

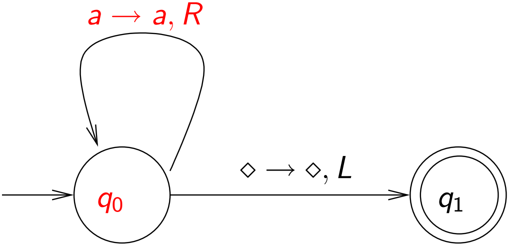
q_0

$T = 1$

Machine de Turing

Exemple de Machine de Turing

Soit la machine suivante qui accepte le langage $L = a^*$:



a	a	a	$\diamond$	$\diamond$	$\diamond$	$\diamond$	$\cdots$
	q_0						

$T = 2$

Rappels

Rappels de complexité

Rappels sur les graphes

Rappels sur la logique

Rappels d'un machine de Turing

Représentation et fonctionnement d'une machine de Turing

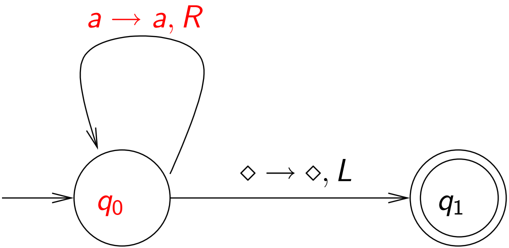
Description formelle d'une machine de Turing

Rappels
Rappels de complexité
Rappels sur les graphes
Rappels sur la logique
Rappels d'un machine de Turing
Représentation et fonctionnement d'une machine de Turing
Description formelle d'une machine de Turing

Machine de Turing

Exemple de Machine de Turing

Soit la machine suivante qui accepte le langage $L = a^*$:



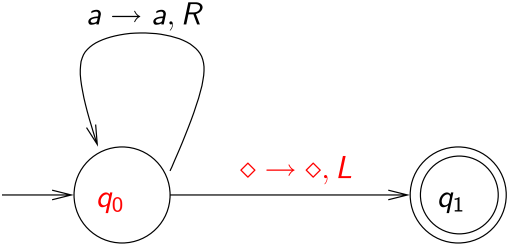
a	a	a	$\diamond$	$\diamond$	$\diamond$	$\diamond$	$\cdots$
		q_0					

$T = 3$

Machine de Turing

Exemple de Machine de Turing

Soit la machine suivante qui accepte le langage $L = a^*$:



a	a	a	$\diamond$	$\diamond$	$\diamond$	$\diamond$	$\cdots$

q_0

$T = 4$

Rappels

Rappels de complexité

Rappels sur les graphes

Rappels sur la logique

Rappels d'un machine de Turing

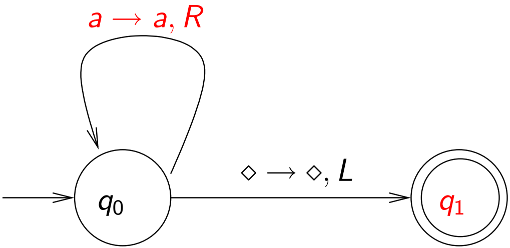
Représentation et fonctionnement d'une machine de Turing

Description formelle d'une machine de Turing

Machine de Turing

Exemple de Machine de Turing

Soit la machine suivante qui accepte le langage $L = a^*$:



a	a	a	$\diamond$	$\diamond$	$\diamond$	$\diamond$	$\cdots$
		q_1					

$T = 5 \cdots$: état final ; arrêt et acceptation.

Rappels
Rappels de complexité
Rappels sur les graphes
Rappels sur la logique
Rappels d'un machine de Turing
Représentation et fonctionnement d'une machine de Turing
Description formelle d'une machine de Turing

Machine de Turing

Rappels

Rappels de complexité

Rappels sur les graphes

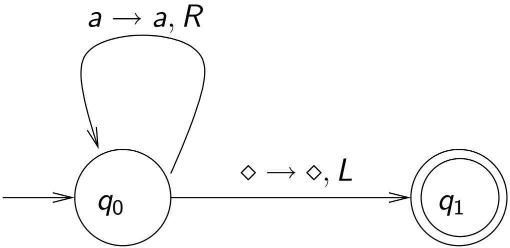
Rappels sur la logique

Rappels d'un machine de Turing

Représentation et fonctionnement d'une machine de Turing

Description formelle d'une machine de Turing

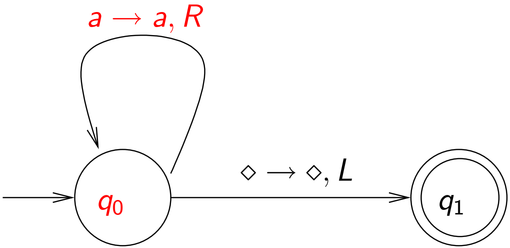
Exemple de rejet



$T = 0$

Machine de Turing

Exemple de rejet



a	b	a	$\diamond$	$\diamond$	$\diamond$	$\diamond$	$\cdots$

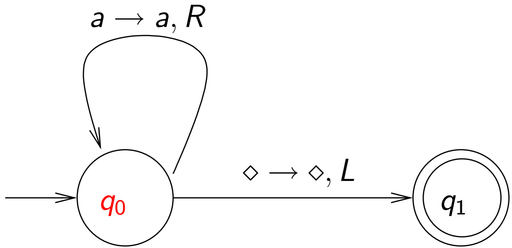
q_0

$T = 1$

Rappels
Rappels de complexité
Rappels sur les graphes
Rappels sur la logique
Rappels d'un machine de Turing
Représentation et fonctionnement d'une machine de Turing
Description formelle d'une machine de Turing

Machine de Turing

Exemple de rejet



a	b	a	$\diamond$	$\diamond$	$\diamond$	$\diamond$	$\cdots$
	q_0						

$T = 2$: pas de transition possible ; arrêt et rejet.

Rappels

Rappels de complexité

Rappels sur les graphes

Rappels sur la logique

Rappels d'un machine de Turing

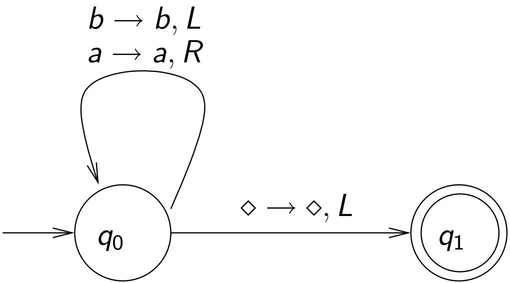
Représentation et fonctionnement d'une machine de Turing

Description formelle d'une machine de Turing

Machine de Turing

Rappels
Rappels de complexité
Rappels sur les graphes
Rappels sur la logique
Rappels d'un machine de Turing
Représentation et fonctionnement d'une machine de Turing
Description formelle d'une machine de Turing

Exemple de boucle infinie

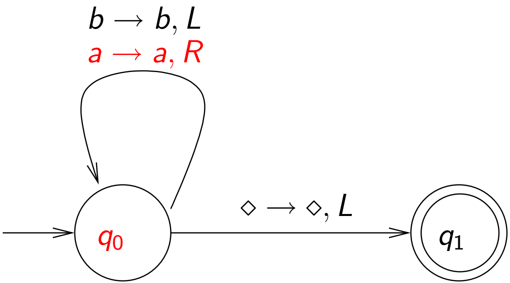


$T = 0$

Machine de Turing

Rappels
Rappels de complexité
Rappels sur les graphes
Rappels sur la logique
Rappels d'un machine de Turing
Représentation et fonctionnement d'une machine de Turing
Description formelle d'une machine de Turing

Exemple de boucle infinie



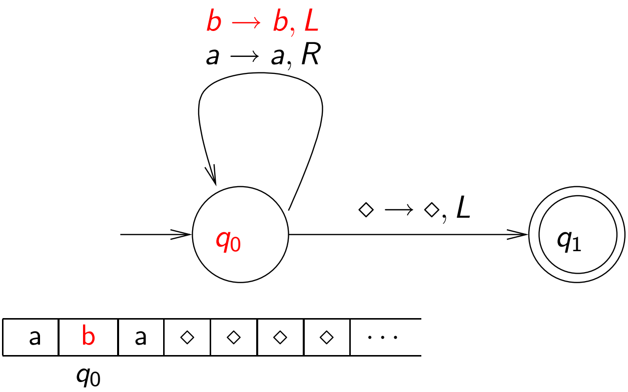
a	b	a	$\diamond$	$\diamond$	$\diamond$	$\diamond$	$\cdots$

q_0

$T = 1$

Machine de Turing

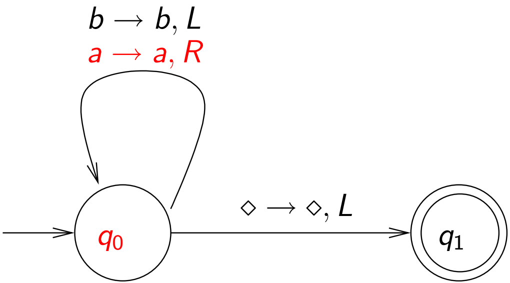
Exemple de boucle infinie



$T = 2$

Machine de Turing

Exemple de boucle infinie



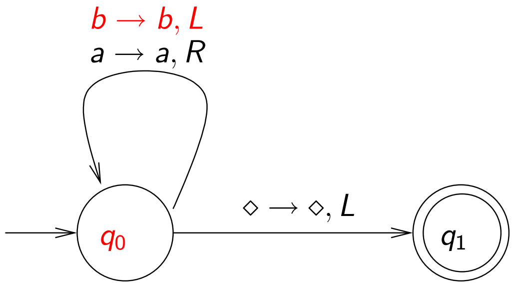
a	b	a	$\diamond$	$\diamond$	$\diamond$	$\diamond$	$\cdots$

q_0

$T = 3$

Machine de Turing

Exemple de boucle infinie



a	b	a	$\diamond$	$\diamond$	$\diamond$	$\diamond$	$\cdots$
	q_0						

$T = 4$

Rappels

Rappels de complexité

Rappels sur les graphes

Rappels sur la logique

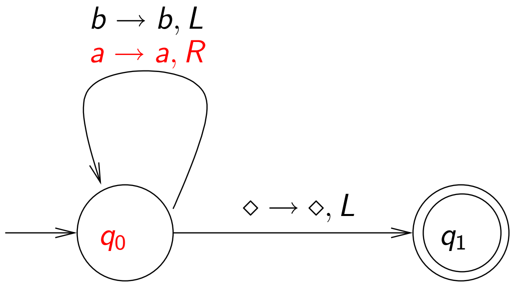
Rappels d'un machine de Turing

Représentation et fonctionnement d'une machine de Turing

Description formelle d'une machine de Turing

Machine de Turing

Exemple de boucle infinie



a	b	a	$\diamond$	$\diamond$	$\diamond$	$\diamond$	$\cdots$

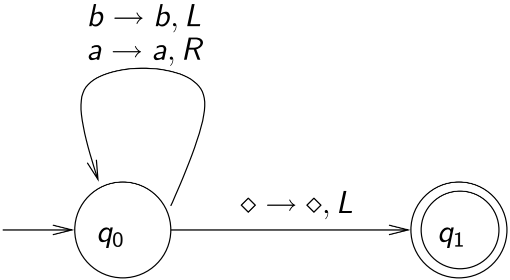
q_0

$T = 5 \cdots$: boucle inifie.

Rappels
Rappels de complexité
Rappels sur les graphes
Rappels sur la logique
Rappels d'un machine de Turing
Représentation et fonctionnement d'une machine de Turing
Description formelle d'une machine de Turing

Machine de Turing

Exemple de boucle infinie



$T =$

Machine de Turing

Définition formelle

$$M = (\mathcal{Q}, \Sigma, \Gamma, \delta, q_0, \diamond, F)$$

avec

- $\mathcal{Q}$: Etats
- Σ : Alphabet d'entrée
- Γ : Alphabet du ruban
- δ : fonction de transition (ex : $\delta(q_1, a) = (q_2, b, R)$)
- q_0 : état initial
- $\diamond$: blanc
- F : Etat final

Rappels
Rappels de complexité
Rappels sur les graphes
Rappels sur la logique
Rappels d'un machine de Turing
Représentation et fonctionnement d'une machine de Turing
Description formelle d'une machine de Turing

Machine de Turing

Rappels
Rappels de complexité
Rappels sur les graphes
Rappels sur la logique
Rappels d'un machine de Turing
Représentation et fonctionnement d'une machine de Turing
Description formelle d'une machine de Turing

Langage accepté

Pour toute machine de Turing M,

$$L(M) = \{w \ : \ q_0 \ w \mapsto^* x_1 \ q_f \ x_2\}$$

avec

- $q_0 \ w$:configuration initiale (état q_0 et tête sur première lettre de w
- $q_1 \ xv \mapsto x \ q_2 v$: déplacement de la tête en lisant la lettre x et en passant de l'état q_1 à l'état q_2
- $\mapsto^*$: occurrence multiple du déplacement $\mapsto$

Machine de Turing

Décidabilité

- Langage décidable : il existe un algorithme qui permet de reconnaître en un temps fini si un mot w appartient ou non à L
- Un langage L est décidé par une machine de Turing M si
  - M accepte L
  - M n'a pas d'exécution infinie