Lycée Les 3 Sources **Bourg-Lès-Valence**
NSI T^le – Thème 4 Année **2024-25**

Les machines de Turing

Présentation

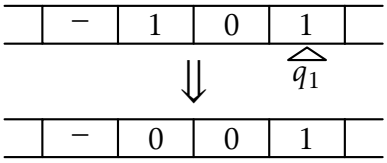
La **machine de Turing** est un modèle théorique, inventé par le mathématicien anglais Alan Turing (1912 – 1954), représentant un programme s'exécutant sur un ordinateur. Son fonctionnement est très "basique". Elle est composée d'une bande infinie sur laquelle elle lit et écrit des symboles d'un alphabet donné (0, 1, ...) et d'un ensemble fini d'états (q_1, q_2, ...).
Le fonctionnement de la machine est défini par un ensemble de règles, appelées **instructions**. Chaque instruction est du type :
Si la machine est dans l'état q et que le symbole S *est lu sur la bande, alors écrire* S' *à la place, se déplacer au symbole de gauche (ou de droite) et passer dans l'état q'.*
Bien que ce fonctionnement soit très simple il permet de réaliser tout ce qu'un ordinateur peut réaliser, quelque soit le langage de programmation utilisé. Par contre, le nombre d'instructions devient vite très grand. Ce modèle sert de base à la définition de la **calculabilité**, qui permet de déterminer ce que peut faire un ordinateur, ou pas.
Il existe de nombreuses variantes de la machine de Turing (alphabet élargi, plusieurs bandes...), mais cela ne donne pas plus de "puissance" à la machine. Pour simplifier la présentation des problèmes, nous rajouterons "–" dans l'alphabet.

Exemple de fonctionnement

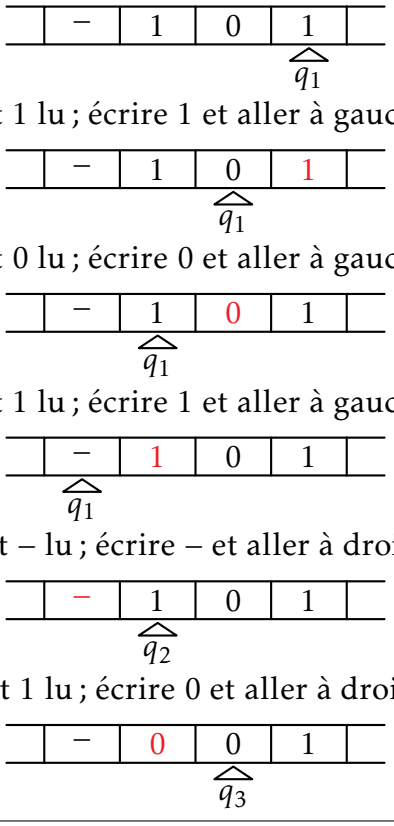
Un programme pour la machine de Turing se présente comme ci-dessous.

État init.	Lu	Écrit	Dir.	État suiv.
q_1	0	0	←	q_1
q_1	1	1	←	q_1
q_1	–	–	→	q_2
q_2	0	1	→	q_3
q_2	1	0	→	q_3

Il sert à résoudre le problème suivant :



Il faut partir de la bande au dessus, avec la tête de lecture qui se trouve dans la position indiquée, dans l'état q_1 et arriver à la bande du bas.
La figure ci-contre montre le déroulement de l'exécution du programme. À la fin, il n'y a plus de règles qui peuvent s'appliquer, donc il s'arrête.

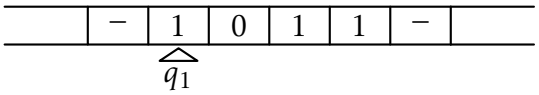


Exercice 1 : Décrire en français le but de ce programme, en supposant qu'il part toujours à droite du texte proposé.

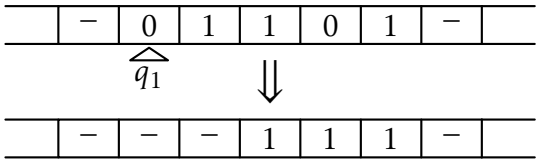
Exercice 2 : On considère le programme suivant.

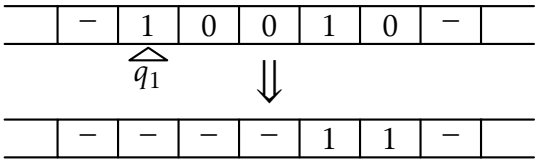
État init.	Lu	Écrit	Dir.	État suiv.
q_1	1	1	$\rightarrow$	q_1
q_1	0	1	$\leftarrow$	q_2
q_2	–	–	$\rightarrow$	q_3
q_2	1	1	$\leftarrow$	q_2
q_3	1	–	$\rightarrow$	q_1

1) Écrire les différentes étapes du ruban pendant l'exécution de ce programme à partir de cette situation initiale.



2) Vérifier que pour les exemples ci-dessous, l'exécution du programme permet de passer du ruban du haut au ruban du bas. On pourra ne représenter que les étapes où il y a un changement d'état.





3) On peut remarquer que l'état q_1 sert à chercher le premier 0 vers la droite, le remplace par un 1 et passe en q_2. De façon similaire, expliciter l'objectif des états q_2 et q_3.
4) Décrire en français ce que fait ce programme.

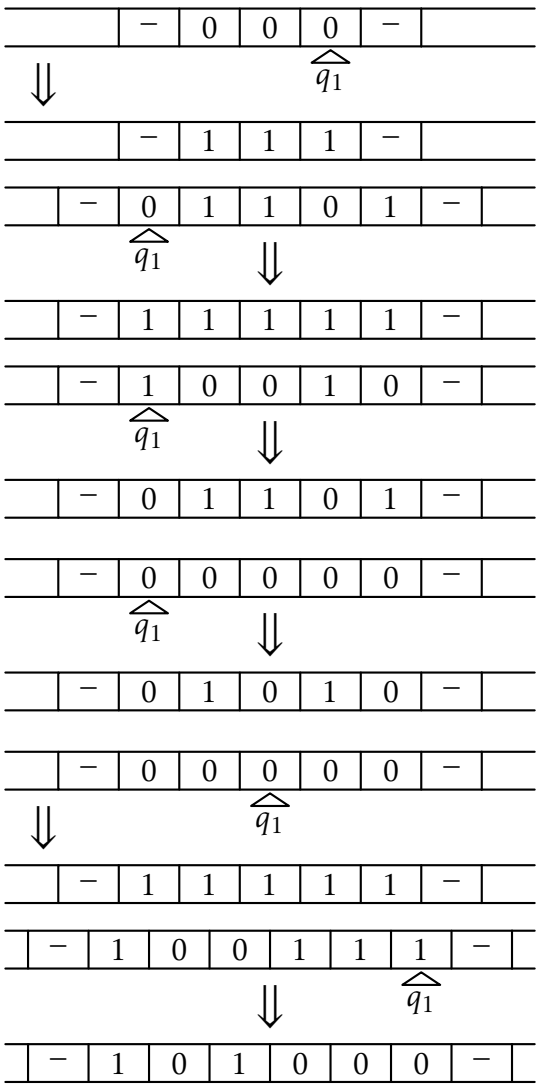
Pour chacun des exercices suivants, la longueur du mot se trouvant sur le ruban initial n'est pas connu à l'avance.

Exercice 3 : Écrire un programme, ne contenant qu'une instruction, permettant d'obtenir l'exécution ci-contre.

Exercice 4 : Écrire un programme, à deux instructions, permettant d'obtenir l'exécution ci-contre.

Exercice 5 : Écrire un programme, à deux instructions, permettant d'obtenir l'exécution ci-contre, où on obtient le complémentaire du mot écrit sur le ruban.

Exercice 6 : Écrire un programme permettant d'obtenir l'exécution ci-contre. Il faut utiliser 2 états. Il faut faire une alternance entre des 0 et des 1.

Exercice 7 : Écrire un programme permettant d'obtenir l'exécution ci-contre.

Exercice 8 : Écrire un programme permettant d'ajouter 1 au nombre écrit en binaire sur le ruban.