TP 7
Architecture des systèmes informatiques – Info3 S5
Machine de Turing

Objectif du TP : Appliquer les principes vus en cours sur la théorie et le fonctionnement de la machine de Turing.

1 Installation du simulateur de machine de Turing

Une machine de Turing est une machine abstraite et universelle capable d'exécuter un programme séquentiel. Elle est composée d'une part d'un ruban séquentiel permettant de stocker les données, muni d'un curseur (ou marguerite ou tête de lecture/écriture) positionné sur le caractère courant, et d'autre part d'un automate à états finis contenant le programme séquentiel à exécuter. Le programme est composé d'un ensemble de transitions entre état. Chaque transition peut réaliser 3 opérations élémentaires sur le ruban :

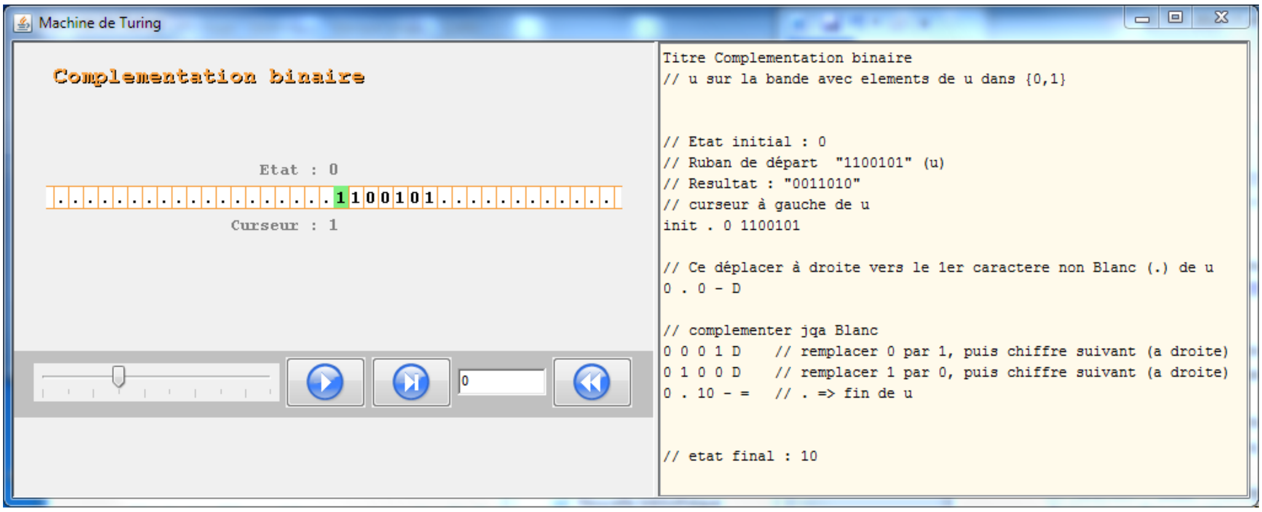
- lire le caractère à la position du curseur
- écrire un nouveau caractère à la position du curseur
- déplacer le curseur à droite ou à gauche

Dans un premier temps, vous installerez le simulateur de Machine de Turing à partir de l'URL : http://www.labri.fr/perso/betrema/MC/Turing.html. Il s'agit d'une archive jar qui peut être exécuté par une machine virtuelle java.

Commencez par lire le manuel d'utilisation de ce simulateur, que vous trouverez à cette même URL.
Vous noterez que chaque transition de la machine est définie par un quintuplet :

Etat courant, caractère lu, état suivant, caractère écrit, déplacement

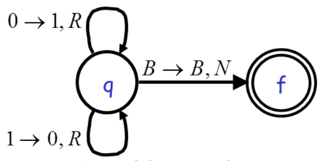
Le caractère « Blanc » est représenté par « . », et « - » désigne n'importe quel caractère. Les déplacements sont désignés par les caractères « D » (à droite), « G » (à gauche), « = » (aucun). Vous pouvez également définir l'état initial de la machine et la valeur initiale du ruban (Init).



2 Première Machine de Turing : la complémentation binaire

`...0011001...` -> `...1100110...`

Reprenez les transparents vus en cours sur la Machine de Turing réalisant la complémentation binaire, et plus particulièrement son graphe de transition. Vous traduirez ce graphe de transition en un programme composé d'une suite de transitions exprimées sous la forme de quintuplets selon la syntaxe du simulateur. Puis vous chargerez ce programme dans le simulateur.



Transition graph

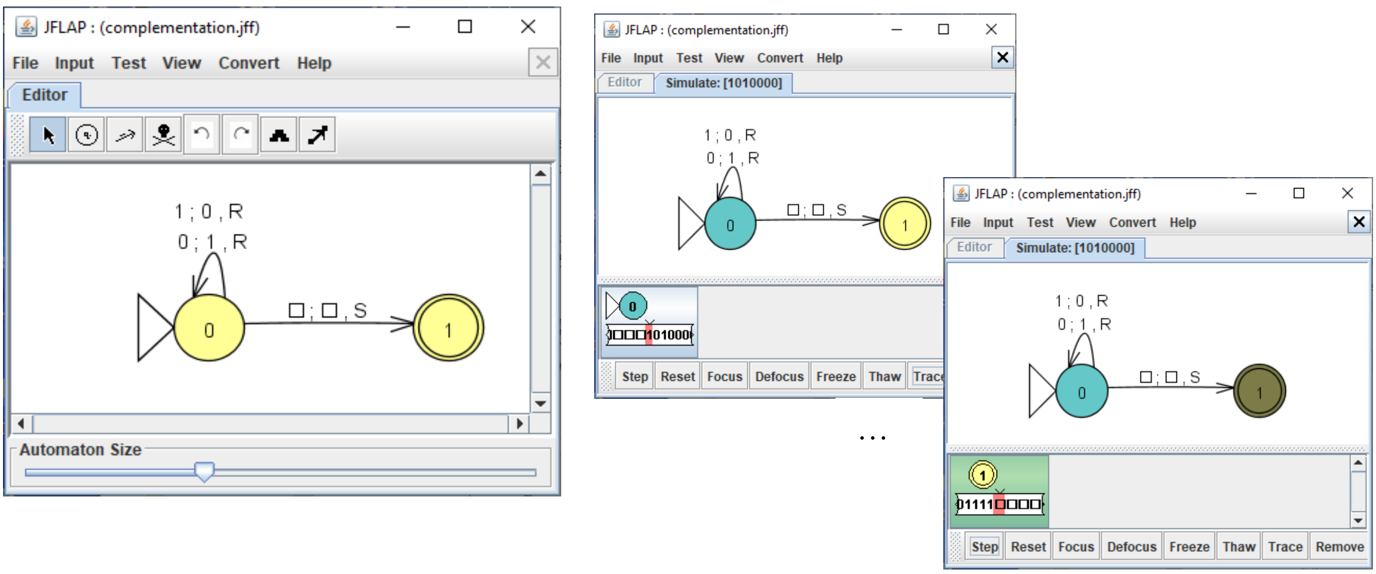
Exécutez cette Machine de Turing en mode pas à pas sur le ruban initialisé à « 0011001 ».
A chaque pas, vous devrez retrouver dans le graphe de transition : la transition activée, l'état courant, et l'état suivant.
Vous conserverez la « trace de l'exécution » sur le ruban, telle que vu en cours.

Cette machine à complémenter fonctionne en parcourant le ruban de gauche à droite. Proposez une nouvelle machine de Turing qui parcourt le ruban en sens inverse (de la droite vers la gauche). Donnez son graphe de transition, puis implémentez cette machine et exécutez-la.

Passage à JFLAP :

Ce premier simulateur a l'avantage d'être très simple. Mais il ne permet pas d'éditer un graphe de transition, ce qui constitue rapidement une limite forte au développement de machines de Turing plus complexes. Dans un deuxième temps, il sera préférable d'utiliser le simulateur JFLAP qui intègre un éditeur de graphe de transition. (JFLAP 7.1 , ou JFLAP 8 beta avec mode trace)

L'exécution de la machine s'effectue via le menu « `Input/Step…` » ou « `Input/Multiple Run (Transducer)` »
Le curseur/marguerite est placé à gauche du ruban au départ (voir le tutoriel JFLAP Machine de Turing).



3 Deuxième Machine de Turing : l'incrémentation binaire

`...010111...` -> `...011000...`

Toujours en reprenant les éléments vus en cours, implémenter la machine de Turing réalisant l'incrémentation d'un nombre binaire.

Pour cela, vous dessinerez son graphe de transition, et en déduirez le programme équivalent à base de quintuplets. Puis vous exécuterez ce programme sur la valeur initiale « 010111 » en vérifiant manuellement les transitions activées dans le graphe. Enfin, vous conserverez la « trace d'exécution » de cette machine.

Reprenez la même procédure afin d'implémenter la décrémentation d'un nombre binaire. Testez votre solution sur la valeur initiale « 110000 »

Reprenez la même procédure pour l'incrémentation d'un nombre en base 10. Testez votre solution sur la valeur initiale « 1999 »

4 Somme de 2 nombres binaires

`...10111+101111...` -> `...1000110...`

En partant les éléments vus en cours, implémenter la machine de Turing réalisant la somme de 2 nombres binaires.

Pour cela, vous dessinerez son graphe de transition, et en déduirez le programme équivalent à base de quintuplets. Puis vous exécuterez ce programme sur la valeur initiale « 10111+101111 » en vérifiant manuellement les transitions activées dans le graphe. Enfin, vous conserverez la « trace d'exécution » de cette machine.

Reprenez la même procédure afin d'implémenter la différence de 2 nombres binaires. Testez votre solution sur la valeur initiale « 100111-1111 »

Reprenez la même procédure pour la somme de 2 nombres en base 10. Testez votre solution sur la valeur initiale « 1235-236 »

Pour aller plus loin, vous pouvez aussi implémenter la machine de Turing qui effectue la somme de 2 nombres binaires comme on le ferait à la main : par sommes binaires successives avec retenue, du poids le plus faible au poids le plus fort. C'est beaucoup plus rapide !

```
  110100     retenue R=(Rn…R1R0)2
   110101    X =(Xn…X1X0)2
+   10110    Y =(Yn…Y1Y0)2
--------
10101011     S , avec (RiSi)2 = Xi+Yi+Ri-1 ' , et R0=0
```

Idée : utiliser des caractères supplémentaires (X Y) pour mémoriser les 0 et 1 des calculs intermédiaires

```
1)...110101+10110..                  5) ...110XYY+10..... (X pour 0, R=1)
2)...11010Y+1011... (Y pour 1, R=0)  ...
3)...11010Y+1011...                  10)..YXXYXYY+
4)...1101YY+101.... (Y pour 1, R=0)  11)..1001011.
```

5 Produit de 2 nombres binaires

L'idée consiste à reproduire la technique de multiplication manuelle en base 2, en transformant le produit par la somme des produits intermédiaires : $X*Y = \Sigma_i \; X*Y_i*2^i$, où $Y = (Y_k Y_{k-1} \ldots Y_1 Y_0)_2$

```
      110101     X
    *  10110     Y = (Y4 Y3 Y2 Y1 Y0)
    --------
     1101010     X*Y1*2^1 avec Y1*2^1 = (10)2
    11010100     X*Y2*2^2 avec Y2*2^2 = (100)2
  1101010000     X*Y4*2^4 avec Y4*2^4 = (10000)2
+-----------
 10010001110     X*Y
```

Ce qui donnera les différentes étapes sur le ruban :

```
    ...110101*10110...
->  ...110101*10110=1101010+11010100+1101010000...
->  ...10010001110
```

Reprenez la même méthodologie que dans les questions précédentes en débutant par la conception du graphe de transition de cette machine de Turing.

Une autre possibilité (plus rapide) est de représenter le produit à partir de décalages binaires (div 2, mul 2 et mod 2) et d'une somme cumulée (voir Technique de multiplication dite russe) :

```
............110101*10110..
                (X * Y)  (53*22=1166)
............1101010*1011..
              (X*2) * (Y/2) , Y mod 2=0, X'=(X*2), Y'=(Y/2) (0 + 106*11)
....1101010+11010100*101..
         X' +   (X'*2) * (Y'//2), Y' mod 2=1, , S=X', X'=(X'*2), Y'=(Y'/2) (106 + 212*5)
..100111110+110101000*10..
       S+X' +    (X'*2) * (Y'//2), Y' mod 2=1, S=S+X', X'=(X'*2), Y'=(Y'/2) (106+212 + 424*2)
..100111110+1101010000*1..
       S+X' +      (X'*2) * (Y'//2), Y' mod 2=0, X'=(X'*2), Y'=(Y'/2) (318 + 848*1)
..10010001110.. S+X', Y' mod 2=1, (1166)
```