Compte Rendu Projet THL

Kichou, Nadir (G7)
nadirkichou@hotmail.fr

Mansoura, Wael (G8)
wael@gmail.com

Ouldali, Najib (G9)
najib@gmail.com

Seddiki, Yacine (G9)
yacine@gmail.com

May 28, 2023

Contents

1 Automate Choisi

Nous avons choisi un automate pour reconnaitre tout les nombres dans $\mathbb{R}$

$$1, 2.0, -4.5, 10, +10, 0.45, -0.78, 85, 10.55\ldots$$

2 Descripton de l'Automate Choisi

2.1 Definition

$$A = (\Sigma, E, e_0, F, \Delta)$$

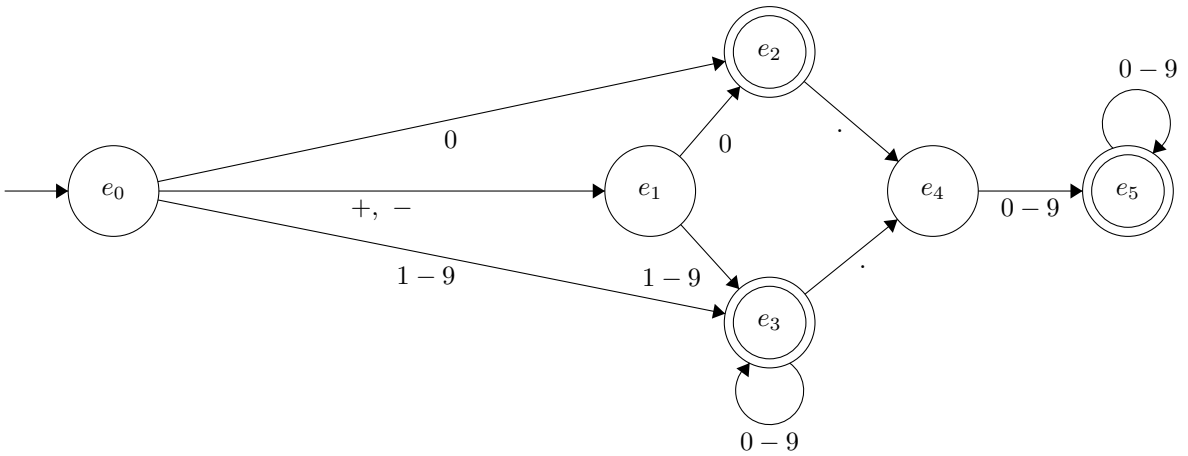
Alphabet: $\Sigma = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, ., -, +\}$
Etats: $E = \{e_0, e_1, e_2, e_3, e_4, e_5\}$
Etats Finaux: $F = \{e_2, e_3, e_5\}$
Etat Initial: $e_0 = e_0$
Transitions:

- $\delta(e_0, +) = e_1$
- $\delta(e_0, -) = e_1$
- $\delta(e_0, 0) = e_2$
- $\delta(e_0, 1) = e_3$
- $\delta(e_0, 2) = e_3$
- $\ldots$

2.2 Representation Matricielle

State	0	1	2	3	4	5	6	7	8	9	.	-	+
e_0	e_2	e_3	e_3	e_3	e_3	e_3	e_3	e_3	e_3	e_3	e_p	e_1	e_1
e_1	e_2	e_3	e_3	e_3	e_3	e_3	e_3	e_3	e_3	e_3	e_p	e_p	e_p
e_2	e_p	e_p	e_p	e_p	e_p	e_p	e_p	e_p	e_p	e_p	e_4	e_p	e_p
e_3	e_3	e_3	e_3	e_3	e_3	e_3	e_3	e_3	e_3	e_3	e_4	e_p	e_p
e_4	e_5	e_5	e_5	e_5	e_5	e_5	e_5	e_5	e_5	e_5	e_p	e_p	e_p
e_5	e_5	e_5	e_5	e_5	e_5	e_5	e_5	e_5	e_5	e_5	e_p	e_p	e_p

2.3 Representation Graphique



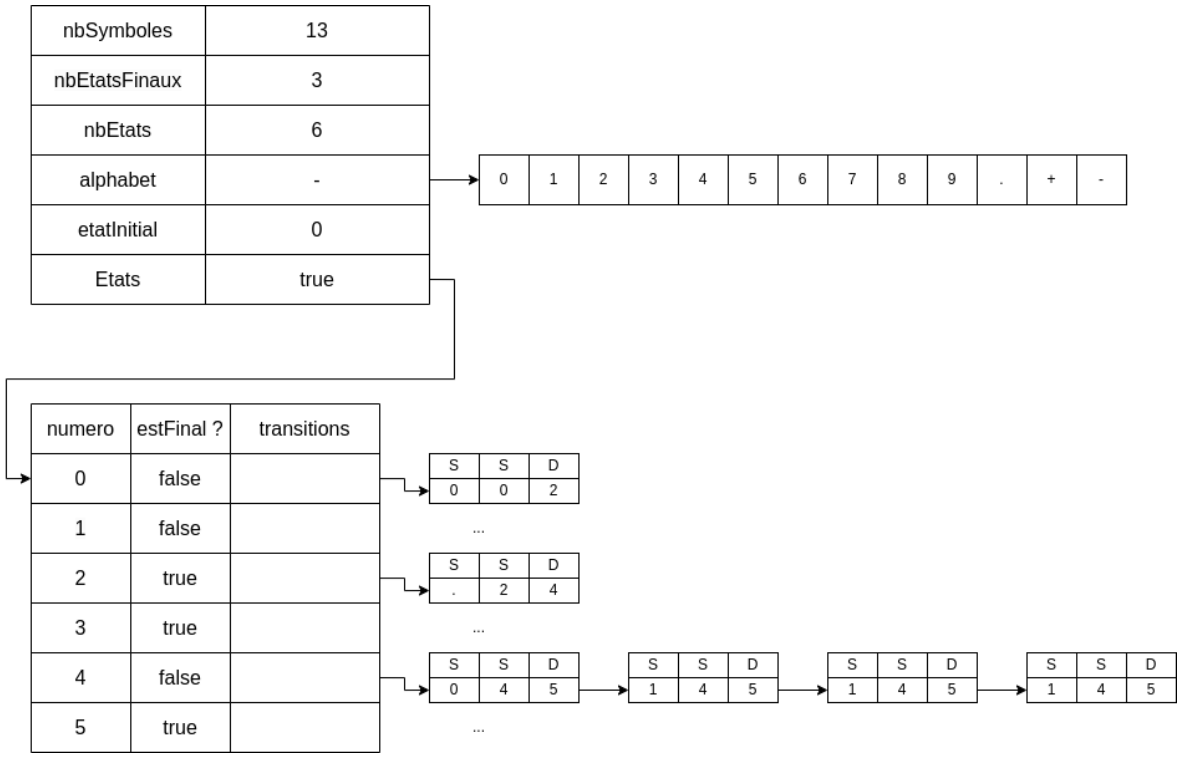
3 Fichier D'Entré

Le fichier d'entré est sous cette forme

```
Nombre Etats = 6
Nombre Symboles = 13
Nombre Etats Finaux = 3
Nombre Transitions = 54
Nombre Mots Test = 6
Etats = 0, 1, 2, 3, 4, 5,
Symboles = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, ., +, -,
Etat Initial = 0
Etats Finaux = 2, 3, 5,
delta(0, +) = 1
delta(0, -) = 1
delta(0, 0) = 2
delta(0, 1) = 3
delta(0, 2) = 3
delta(0, 3) = 3
delta(0, 4) = 3
delta(0, 5) = 3
delta(0, 6) = 3
delta(0, 7) = 3
delta(0, 8) = 3
delta(0, 9) = 3
delta(1, 0) = 2
delta(1, 1) = 3
delta(1, 2) = 3
delta(1, 3) = 3
delta(1, 4) = 3
delta(1, 5) = 3
delta(1, 6) = 3
delta(1, 7) = 3
delta(1, 8) = 3
delta(1, 9) = 3
delta(3, 0) = 3
delta(3, 1) = 3
delta(3, 2) = 3
delta(3, 3) = 3
delta(3, 4) = 3
delta(3, 5) = 3
delta(3, 6) = 3
delta(3, 7) = 3
delta(3, 8) = 3
delta(3, 9) = 3
delta(2, .) = 4
delta(3, .) = 4
delta(4, 0) = 5
delta(4, 1) = 5
delta(4, 2) = 5
delta(4, 3) = 5
delta(4, 4) = 5
delta(4, 5) = 5
delta(4, 6) = 5
delta(4, 7) = 5
delta(4, 8) = 5
delta(4, 9) = 5
delta(5, 0) = 5
delta(5, 1) = 5
delta(5, 2) = 5
delta(5, 3) = 5
delta(5, 4) = 5
delta(5, 5) = 5
delta(5, 6) = 5
delta(5, 7) = 5
delta(5, 8) = 5
delta(5, 9) = 5
Mots Test Valide = -40.00, 480790, -10, 8, 4, +2,
Mots Test Invalide = 007.07, -.1., +-+78549, --+4.887, -+-.15.+2, -45.45
```

4 Structures de Données

Pour stocker les noms des etats et les identifier nous avons utilisé des entiers au lieu de chaines de characteres pour un soucis de simplicité




```
{
  int nbSymboles;
  int nbEtats;
  int nbEtatsFinaux;
  char *alphabet;
  // TODO: change to string ?
  int etatInitial;
  Etat **etats;
};
```

Listing 1: Structure Automate

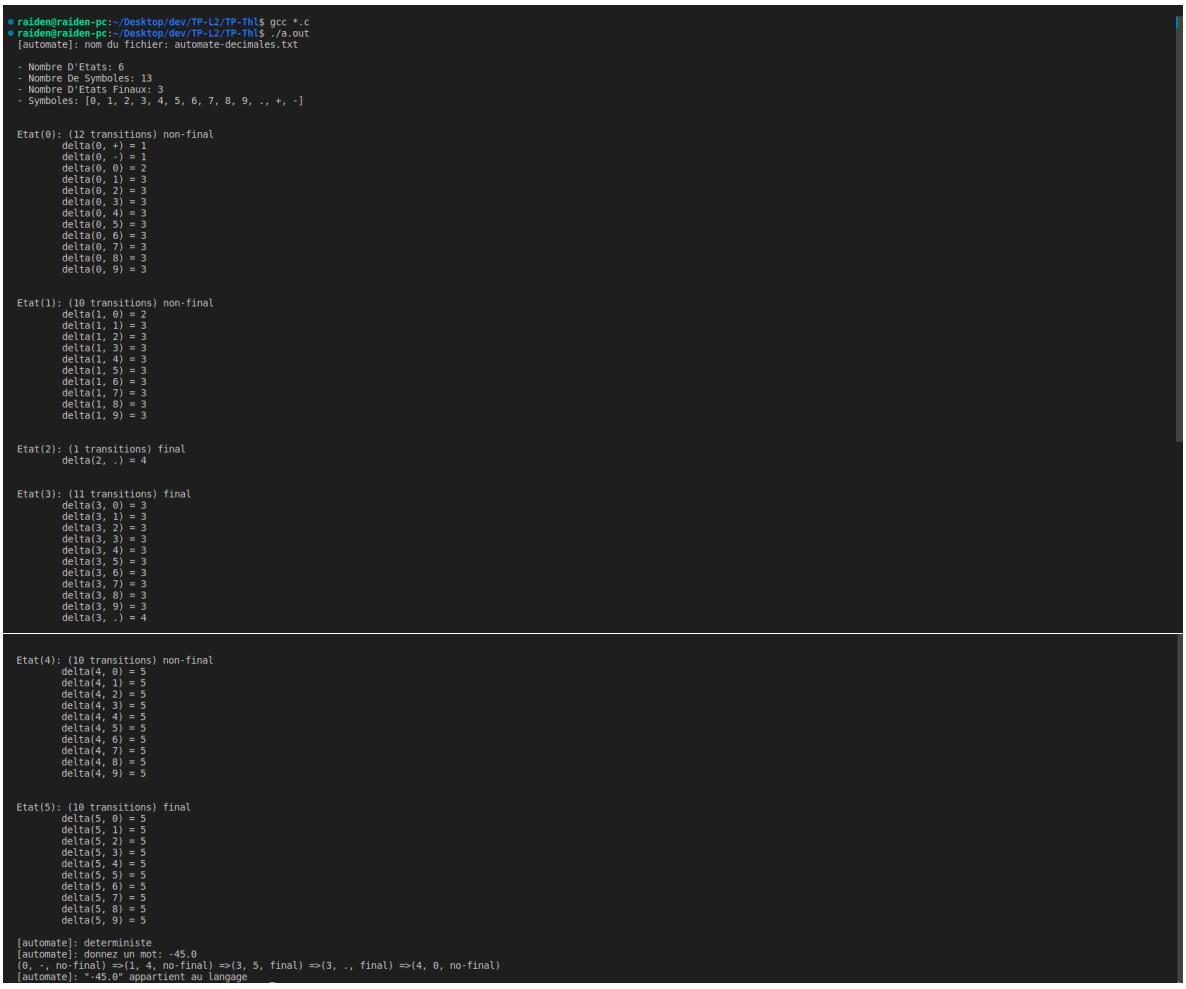
```
struct Etat
{
  // TODO: change to string
  int numero;
  bool estFinal;
  List *transitions;
};
```

Listing 2: Structure Etat

```
struct Transition
{
  char symbole;
  // TODO: change to string
  int source;
  // TODO: change to string
  int destination;
};
```

Listing 3: Structure Transition

5 Captures D'Ecran de L'execution



6 Autre

Si vous souhaitez consulter le code du projet vous pourrez le retrouver sur mon github.