TP COMPILATION — Une calculette

On s'intéresse à un petit langage d'expressions, appelé LACA, qui permet de décrire des suites d'expressions. Les expressions sont constituées d'opérateurs infixés binaires (-, *, +), de constantes entières et sont terminées par le signe =. La suite d'expressions se terminent par l'expression réduite à =. Les constantes sont représentées par une suite de caractères consécutifs représentant des chiffres. Les caractères séparateurs sont le caractère blanc et le retour à la ligne.

La valeur d'une expression est obtenue par une évaluation de gauche à droite des opérateurs.

Un exemple d'entrée :

```
2 + 34  *2     =
5=
=
```

▷ **Question 1** *Définir la syntaxe à partir du vocabulaire terminal {NUM, PLUS, MOINS, MULT, EGAL, FF}, où FF désigne la fin de fichier. Donner un automate de reconnaissance de ce langage.*

▷ **Question 2** *Définir la lexicographie, i.e la représentation des terminaux NUM, PLUS, MOINS, MULT, EGAL, FF. Donner l'automate associé à la lexicographie.*

▷ **Question 3** *Ecrire un programme permettant de reconnaître ce langage et, le cas échéant, de signaler les erreurs.*

▷ **Question 4** *Compléter le programme ci-dessus pour ajouter l'évaluation des expressions.*

▷ **Question 5** *On ajoute dans les expressions la notation $i qui représente le résultat de la ième expression. $i ne peut apparaître dans l'expression de numéro j que si i<j. Etendre la lexicographie et la syntaxe en conséquence. Compléter ensuite le programme de la question 4.*