

---

---

## TD N°1

---

---

### Exercice 01

Définir les expressions régulières pour les langages ci-dessous :

1. Les mots composés de lettres en minuscule de l'alphabet et contenant exactement les chiffres 1, 2, 3 et 4 toujours présents dans l'ordre croissant : 1234, 1ab2c34, abc12f3ad4, ...
2. Les mots non vides commençant par une lettre minuscule de l'alphabet, et chaque lettre est toujours suivie d'un chiffre. Ces mots sont de longueur paire :  
e1, b2c1d3, ... sont des mots acceptés,  
e, 1, da1d, abcd, ... ne sont pas acceptés.
3. Les mots non vides, composés de caractères minuscules de l'alphabet, mots où le caractère 'e' est soit complètement absent ou soit présent toujours par paire :  
abc, ee, beecce, eeedaf, ... sont des mots acceptés,  
e, bec, deeea, ... ne sont pas acceptés.
4. Les mots non vides, composés de lettres minuscules ou majuscules de l'alphabet, ne contenant au plus qu'un seul 'e' ou 'E' à la fois :  
E, ea, aabCDfg, ... sont des mots acceptés,  
aebE, ab12de, abcdEfeF, abecEdae, ... ne sont pas acceptés.
5. Les mots composés de lettres minuscules de l'alphabet,  
– où 'a', 'b' et 'c' apparaissent au plus une fois (c'est-à-dire 0 ou 1 fois),  
– si 'a' est présent, il n'y aura pas 'b' ni 'c' à sa gauche de près ou de loin,  
– si 'b' est présent, il n'y aura pas 'c' à sa gauche de près ou de loin.  
Par exemple :  
a, b, c, abde, btgcf, xaxbefc, ... sont des mots acceptés, aeaf, ebdag, acbde, ... ne sont pas acceptés.

### Exercice 02

1. Donner une expression régulière étendue permettant de reconnaître des entiers naturels :
  - sans zéro non significatif à gauche,
  - composés de nombre pair de chiffres pairs,
  - ou composés de nombre impair de chiffres impairs.

**Exemples d'entiers acceptés:** 224466 246820 135 9975311

**Exemples d'entiers non acceptés:** 024666 2410 1357 13257

2. Donner une expression rationnelle étendue permettant de reconnaître des durées (strictement inférieures à 24h) en heures, minutes et secondes respectant les contraintes suivantes :
  - Une durée s'exprime avec le format --h--m--s.
  - Les secondes et minutes sont comprises entre 0 et 59 et composées d'un ou deux chiffres.
  - Les heures sont comprises entre 0 et 23 et composées d'un ou deux chiffres.
  - Les heures ne peuvent être présentes que si les minutes y sont et les minutes ne peuvent être présentes que si les secondes y sont.

**Exemples de durées acceptées :** 05s 05m00s 02h3m04s 2h03m4s

**Exemples de durées non acceptées:** 05m : sans la présence des secondes

3. Donner une expression rationnelle étendue en Lex permettant de reconnaître les numéros d'immatriculation composés de :
  - 3 à 5 chiffres ne commençant par le chiffre 0,
  - suivis de 2 lettres majuscules en excluant WW,
  - terminant par 2 chiffres en excluant 00

**Exemples de numéros acceptés :** 123WA01 64725AA45

4. Donner une expression régulière étendue permettant de reconnaître les adresses IP v4, (composé de 4 champs séparés par des points. Chaque champ appartient à l'intervalle [0,255] :
  - La première adresse c'est 0.0.0.0
  - Et l'adresse maximale c'est 255.255.255.255

### Exercice 03

Donner un fichier d'entrée pour Lex permettant

- de lire, jusqu'à la fin de la source d'entrée, les nombres entiers positifs écrits en octal, en décimal ou en hexadécimal, les nombres flottants,
- de les afficher avec une indication du type du nombre lu (octal, décimal, hexadécimal ou flottant),
- de compter le nombre de ligne tapé par l'utilisateur,
- d'afficher à la fin le nombre total de lignes.

### Exercice 04

On souhaite utiliser Lex pour analyser le langage de commandes suivant :

- une commande est composée d'un nom de commande, suivi d'une liste optionnelle d'arguments, suivie d'une liste facultative d'options ;
- une liste d'arguments est une suite d'arguments ;
- une liste d'options est une suite non vide d'options encadrée par [ et ], à l'intérieur de laquelle les options sont séparées par virgule « , » ;
- une option est un caractère précédé d'un tiret ;
- un argument est un identificateur, de même qu'un nom de commande.

**Exemple :**

macom arg1 arg2 [-a,-b] est une commande, ainsi que macom [-f] et macom.

### Exercice 05

Nous considérerons dans la suite qu'un tableau en html est défini comme suit:

- Le tableau est encadré par les balises <TABLE> et </TABLE>.
- Chaque ligne est encadrée par les balises <TR> et </TR>
- Les cellules de valeur sont encadrées par les balises <TD> et </TD>

Exemple : le code html ci-dessous

```
<TABLE>
<TR>
<TD> Valeur 11 </TD>
<TD> Valeur 12 </TD>
<TD> Valeur 13 </TD>
</TR>
<TR>
<TD> Valeur 11 </TD>
<TD> Valeur 12 </TD>
<TD> Valeur 13 </TD>
</TR>
</TABLE>
```

Certaines parties du fichier peuvent être des commentaires. Les commentaires html sont encadrés par <!-- et -->.

Ecrivez un programme **Lex** qui prend en entrée un programme HTML que l'on supposera bien structuré et qui pour chaque tableau, écrit la phrase :

*"le tableau commence à la ligne i et contient k lignes"*

Où *i* est le numéro de la ligne du code HTML où commence le tableau et *k* est le nombre de lignes dans ce tableau (pour l'exemple précédent,  $k = 2$ ). Cependant vous ne devrez pas comptabiliser les lignes en commentaires, ni produire un affichage pour des tableaux qui seraient entièrement en commentaires.