

TP2 : Lex er Yacc

November 2021

1 Test

Dans TP2/expression

Créer un fichier *expression.l* et *expression.y* avec le contenu du mail.

Compiler *expression.l* et *expression.y* :

```
bison -d expression.y
```

```
flex expression.l
```

```
gcc -o expression expression.tab.c lex.yy.c -ll
```

```
./expression
```

2 Flex et bison

Q1. Écrire un programme flex et bison qui reconnaît les mots $a^n b^n \forall n \in N$.

Q2. Écrire un programme flex et bison qui reconnaît un nombre (sur l'alphabet $\{0, 1, 2, 3\}$) qui est un palindrome avec un unique 0 au milieu:

1230321 Oui

123321 Non

12303321 Non

Q3. Écrire un programme flex et bison qui reconnaît si dans un mot le nombre de a est égal au nombre de b. Le mot ne doit contenir que des a et des b.

Faire la même chose dans un fichier flex.

3 Bison et Précédence

Reprendre l'exemple *expression*:

Q1. Dans l'exemple *expression*, on accepte les divisions pas 0, modifier l'exemple pour un afficher un message d'erreur pour division par 0.

Q2. Modifier l'exemple *expression* pour qu'il n'y est plus de priorité entre les opérateurs $+ - */$:
 $1 + 2 * 5 + 6/3$ donnera 7 au lieu de 13.

Q3. Modifier l'exemple *expression* pour que l'opérateur multiplicateur soit prioritaire sur le diviseur.

$2 * 8/2 * 2$ donnera 4 au lieu de 16.

Q4. Modifier l'exemple *expression* pour qu'il n'y est plus de priorité entre les opérateurs + - */ et l'évaluation de l'expression est faite de gauche a droite
 $1 + 2 * 5 + 6/3$ donnera 15 au lieu de 13.

4 Flex et bison - Logique

Écrire un fichier flex *logic.l* et bison *logic.y* qui reconnaît une expression logique et l'évalue (si elle est valide). On utilisera comme mot-clé : Vrai, Faux, non, et, ou, ainsi que les parenthèses ouvrantes (et fermentes). Pour l'évaluation de l'expression, il faudra tenir compte des priorités sur les opérateurs (non, puis et, puis ou) et des priorités données par les parenthèses.

Exemple :

```
Nicolass-MacBook-Air:logic nicolasfayard$ ./logic
Vrai et ou Faux
syntax error
Nicolass-MacBook-Air:logic nicolasfayard$ ./logic
Vrau et Vrai
Vrausyntax error
Nicolass-MacBook-Air:logic nicolasfayard$ █
Nicolass-MacBook-Air:logic nicolasfayard$ ./logic
Vrai ou Faux et Faux
Vrai
(Vrai ou Faux) ou Faux
Vrai
non (Vrai ou Faux) ou Faux
Faux
(Vrai ou Faux) et Vrai
Vrai
(Vrai ou Faux) et non Vrai
Faux
non (non Vrai et Vrai) ou Faux et Vrai
Vrai
```