

# TP3 Yacc

## 1 Musique

Sur Plubel, l'archive « `notesLex.zip` » associée au TP3 permet de construire un analyseur syntaxique (Lex) qui détecte des couples {duree note} comme dans le TD6.

Rappel du TD 6 : on veut développer un analyseur Lex Yacc qui permette de décoder un fichier de notes de musique sous la forme :

```
duree note duree note ...
```

Il faut au moins une paire {duree note} dans le fichier. La durée est décrite par {R, B, N}, la note par {do, re, mi, fa, sol, la, si}. On associe des valeurs numériques aux durées {4, 2, 1} pour {R, B, N} respectivement. On veut calculer la durée totale des notes.

1. utiliser la grammaire écrite en TD et le fichier Lex fourni pour construire un analyseur Yacc qui permette la durée totale des durées.

```
Rdo Bfa Nsol -> 7 temps
```

Compilation :

```
yacc notes.yacc
yacc -d notes.yacc
lex notes.lex
gcc -Wall -c lex.yy.c
gcc -Wall y.tab.c lex.yy.o -lfl -o analyseur
```

2. on veut maintenant ajouter des nombres devant chaque paire {duree note} pour tenir compte d'une répétition, par exemple : "2Rdo" correspond à "RdoRdo". Ce nombre n'est pas indispensable (s'il n'y a pas de répétition). Utiliser une %union de `yyval` pour transmettre un texte (la note) ou un entier (répétition ou durée) et l'afficher en développant :

```
2Rdo Bfa 4Nsol -> Rdo Rdo Bfa Nsol Nsol Nsol Nsol : 14 temps
```

## 2 Application au projet

Écrire l'analyseur Yacc du projet, à partir de la grammaire donnée sur Plubel (*cf* la correction de l'étape 3).