

TD Lex

Yves Stadler

30 mars 2009

Exercice numéro 1 : Steam group statistics

Supposons que l'on dispose du fichier suivant provenant d'une page d'un groupe steam.

```
Steam Rating:4.1
Playing time:92.2 hrs this week
Member Avg:13.2 hrs past two weeks
```

```
Left for Dead
30 hrs
```

```
World of Goo
2.1 hrs
```

```
Mirror's Edge
1.2 hrs
```

Ecrire un fichier source Lex qui va parser le fichier et afficher : Le groupe à un ratio de 4,1. Les joueurs ont joué 92.2 heures au total cette semaine. La moyenne par joueur est de 13,3 heures de jeu par jour sur deux semaines. Le groupe à passé 30 heures sur Left4Dead. Le groupe à passé 30 heures sur World of Goo. Le groupe à passé 1,2 heures sur Mirror's Edge.

On pensera à utiliser un entier status. Si status == 0 alors on a rien lu encore. Si status == 1 on a lu un premier nombre. Si status == 2 on a lu un second nombre, ... etc.

- Commencez par écrire une expression régulière qui va représenter un nombre flottant
- Écrivez ensuite la partie de définition du script lex (N'oubliez pas d'importer les librairies standards, et de définir et d'initialiser status)
- Écrivez ensuite la partie des productions. On supposera que le fichier a toujours le même format que celui présenté ci-dessus. Ainsi le premier nombre correspondra au ratio, le second au temps de jeu total, etc...

Solution

Faire attention :

- pour utiliser une notion que l'on vient de définir, il faut utiliser les accolades.

- on peut utiliser une notion pour définir autre
- la variable status permet de compter le nombre de flottant déjà vus. On utilise un switch pour faire les affichages
- la dernière productions permet de ne pas afficher les lignes où aucun flottant n'est reconnu.

```
%{
#include "stdio.h"
int status;
status = 0;
}%
%option noyywrap

nombre          [0-9]+
flottant         {nombre}(\.{nombre})?
%%

{flottant}      {
switch(status) {
    case 0: printf("Le groupe à un ratio de %s", yytext);
             status++;
             break;
    case 1: printf("Les joueurs ont joué %s heures au total cette semaine.", yytext);
             status++;
             break;
    case 2: printf("La moyenne par joueur est de %s heures de jeu par jour
sur deux semaines.", yytext);
             status++;
             break;
    case 3: printf("Le groupe à passé %s heures sur Left4Dead.", yytext);
             status++;
             break;
    case 4: printf("Le groupe à passé %s heures sur World of Goo.", yytext);
             status++;
             break;
    case 5: printf("Le groupe à passé %s heures sur Mirror's Edge.", yytext);
             status++;
             break;
}
}

. {}

%%
int main(void) {
    yylex();
    return 0;
}
```

Exercice numéro 2 : elle met du vieux bash sur son balcon

Écrire un script bash qui dit si le fichier passer en paramètre est une archive tar, tgz, ou bzip2. Hint, il y a plusieurs **cas**.

```
echo -n "Affichage du contenu de l'archive $1 "  
case $1 in  
  *.tar)  
    echo "(tar compresse)"  
    ;;  
  *.tgz)  
    echo "(tar compresse gzip)"  
    ;;  
  *.bz2)  
    echo "(tar compresse bzip2)"  
    ;;  
  *)  
    echo "Erreur, ce n'est pas une archive"  
    ;;  
esac
```