

Sous-titrage de films en jflex

L'objectif de ce TP est d'écrire quelques programmes aidant à la manipulation de fichiers de sous-titres de films. Un tel fichier contient la suite des sous-titres d'un film, avec des indications sur le moment où chaque sous-titre doit s'afficher et sa durée d'affichage. Il existe plusieurs formats pour ce type de données. Nous nous intéresserons au format .SUB.

Dans un fichiers .SUB, chaque sous-titre est défini sur une seule ligne. En début de ligne on trouve le moment de début d'affichage du sous-titre (en nombre d'images depuis le début du film) entre accolades, puis le temps de fin d'affichage entre accolades, enfin le texte du sous-titre. Dans ce texte, un séparateur vertical | indique un saut de ligne.

Voici un exemple de fichier .SUB :

```
{1138}{1260}La gagnante de la bourse|d'études de 30000 dollars...  
{1262}{1369}est Miss Louisiane, Erika Schwarz.  
{1371}{1430}Et la nouvelle Miss Amérique...  
{1433}{1536}est Miss Kansas, Tara Dawn Holland!
```

Les squelettes et fichiers de test utilisés dans ce TP sont disponibles dans le répertoire /public/pil/tp2 sur les machines du 336 ainsi que sur le site ecampus. Vous disposez d'un fichier test.SUB qui vous permet de tester vos programmes. On rappelle la suite de commandes à utiliser pour tester vos programmes :

```
$ jflex sous-titre.jflex  
$ java Lexer.java  
$ java Lexer test.SUB
```

Travail à effectuer

a) Prise en main

Pour commencer, on ne va pas traiter dans les sous-titres le caractère | pour le passage à la ligne qui sera donc vu comme un caractère du texte, (on les gèrera par la suite).

Utilisez le squelette `sous-titre-a.jflex`.

Votre premier programme jflex consistera à lire le fichier et à le réafficher dans un format vous permettant de vérifier que cette lecture est correcte. Sur un fichier contenant la ligne :
{1138}{1260}La gagnante de la bourse d'études de 30000 dollars... cela doit afficher sur une seule ligne :

```
Du temps {1138} jusqu'au temps {1260} le sous titre: La gagnante de la bourse  
d'études|de 30 000 dollars...
```

Question 1 : Proposez une solution pour distinguer le temps du début du temps de fin d'affichage sachant qu'ils correspondent tous les deux à la même expression régulière ?

Question 2 : Ecrivez le programme.

Indication : le contenu d'un sous-titre est une chaîne de caractères qui ne commence pas par "}" et qui ne contient pas de retour à la ligne.

Question 3 : A présent, on vous demande de ranger les temps dans des variables de type int, car on devra les traiter comme tels dans la suite. De plus, on doit les mémoriser car on devra calculer des durées. On utilisera un tableau de deux entiers, le premier étant le temps de début,

et le deuxième le temps de fin d’affichage.

Indications : il vous faudra utiliser la méthode `substring` de la bibliothèque `string`, pour extraire la sous-chaine qui contient juste la suite des chiffres sans les accolades. Cette méthode prend deux paramètres entiers qui sont les index où commence et termine la sous-chaine. Il vous faudra aussi utiliser la méthode statique `Integer.parseInt` qui prend en argument une chaîne de caractères, qui doit contenir seulement des chiffres, et construit l’entier associé.

b) Le cas des sous titres sur plusieurs lignes

Généralisez ce premier programme d’affichage pour prendre également en compte les sauts de lignes (matérialisés par le caractère `\n`). On remplacera l’expression `SUBTITLE` précédente par deux expressions rationnelles différentes : `FSTLINE` qui identifie la première ligne et `LINE` qui identifie les éventuelles autres lignes de sous titres, avec les deux actions suivantes :

```
{FSTLINE} { System.out.println( "soustitre :" + yytext() );}  
{LINE} { System.out.println( yytext() );}
```

c) Affichage avec timing fixe

On ignore à nouveau le caractère `|` de passage à la ligne dans les sous-titres. Le programme suivant est destiné à afficher les sous-titres pendant 2 secondes, avec une seconde de blanc. La partie contenant le nombre d’images est ignorée. Il reste juste à compléter les expressions régulières comme dans les questions précédentes, et à ajouter les actions associées. Vous utiliserez le squelette `sous-titre-c.jflex` qui contient les fonctions `clearConsole()` et `pause()`.

d) Affichage avec le bon timing.

A présent, modifiez le programme pour tenir compte du timing spécifique du film. Le sous-titre doit rester affiché exactement le temps spécifié dans le fichier `.sub`. Attention, ce temps est compté en images et pas en secondes. Par exemple, pour les sous titres

```
{1138}{1260}La gagnante de la bourse|d’études de 30 000 dollars...  
{1300}{1369}est Miss Louisiane, Erika Schwarz.
```

Le premier sous-titre doit s’afficher durant $1260 - 1138$ images soit 22 images qui correspondent à $22000/24$ millisecondes. Il y a ensuite une pause entre les images 1260 et 1300 soit $40000/24$ millisecondes.

Le traitement se fera “à la volée” en utilisant uniquement les deux variables d’entiers $t[0]$ et $t[1]$ et en adaptant la fonction `pause` pour qu’elle traite un nombre d’images.

e) Pour aller plus loin

Les fichiers de sous-titres les plus répandus sont plutôt au format SRT.

Chaque sous-titre se présente ainsi

```
1  
00:00:24,108 --> 00:00:27,194  
<i>Il y a un pré  
dans mon univers idéal</i>
```

Avec un index de sous-titre (de 1 à n) s’il y a n sous-titres sur une seule ligne, un intervalle de temps de format `hh : mm : ss,ms` avec le début et la fin séparés par le symbole `-->` puis la partie sous-titre sur une ou plusieurs lignes et pouvant contenir des balises html.

Reprenez les questions du TP sur ce nouveau format de sous-titre (on pourra juste ignorer les balises html en les supprimant). Quelles difficultés supplémentaires pose ce format ? Comment pouvez-vous les résoudre ?