

# L3 informatique, 2122-2023, Projet de LFC

## Traduction de iCal en html

### ETAPE 5, à rendre à la fin de votre dernier TP

---

#### Corrigé de l'étape 4

Les fichiers lex et yacc que vous deviez obtenir à la fin de l'étape 4 sont en ligne sur plubel.

Rappel : Le TP2 doit être terminé. Si ce n'est pas le cas, il est à terminer en même temps que l'étape 5.

#### Etape 5 : Analyse sémantique et traduction

Vous devez ajouter les actions et fonctions nécessaires pour effectuer les traitements suivants.

##### *Complétion de la table des symboles*

- Pour chaque date enregistrée dans la table, déterminer et enregistrer s'il s'agit d'une date de début ou de fin d'événement.
- Pour chaque texte enregistré dans la table, déterminer et enregistrer s'il s'agit d'un titre, d'un lieu ou d'une description d'événement.

##### *Analyse sémantique*

- Vérifier la validité de chaque date.
- Pour chaque événement, vérifier que la date de fin est bien postérieure à la date de début.
- Pour chaque événement répétitif avec date limite, vérifier que la date limite est bien postérieure à la date de début.
- Pour chaque événement, vérifier qu'il n'y a pas plusieurs alarmes en même temps.

Remarque : Il peut être nécessaire que, pour certaines unités lexicales, l'analyseur lexical passe en même temps une valeur à l'analyseur syntaxique. Celle-ci peut être simplement l'indice dans la table des symboles auquel est stocké le mot reconnu comme appartenant à cette unité lexicale.

Par exemple, si on reprend l'exemple du TP2, lorsque l'analyseur lexical lit la date de début d'événement, il stocke les informations concernant cette date à l'indice 1 de la table des symboles et envoie l'unité lexicale DATEVTU à l'analyseur syntaxique. S'il envoie comme attribut de l'unité lexicale DATEVTU l'indice auquel est stocké la date dans la table (soit 1), l'analyseur syntaxique pourra ainsi accéder aux informations déjà stockées pour cette date et compléter les informations manquantes.

##### *Traduction en html*

A partir de la table des symboles, créer un fichier html qui affichera tous les événements selon la présentation suivante :

- Les événements uniques apparaissent en noir, les événements répétitifs en vert et les événements à la journée en rouge.
- Pour chaque événement :
  - Ligne 1 : les dates et heures de début et de fin, en Titre1, sous la forme  
**Du date\_et\_heure\_début au date\_et\_heure\_fin**  
Pour les événements à la journée, on doit juste avoir  
**Le date, toute la journée**
  - Ligne 2 : le titre, en Titre2, de l'événement.  
Si le titre est vide, écrire le texte « Sans titre » à la place.

- Ligne 3 : le lieu, précédé du texte « Lieu : », en « normal ».  
Si le lieu est vide, écrivez le texte « Pas de lieu » à la place.
- Ligne 4 : la description, précédée du texte « Description : », en normal.  
Si la description est vide, écrivez le texte « Pas de description » à la place.
- Ligne suivante : les informations de répétition pour les événements répétitifs, en normal, sous la forme :

**Doit se répéter chaque** *fréquence*, **les** *liste\_jours*, *nombre\_de\_fois fois*

ou

**Doit se répéter chaque** *fréquence*, **jusqu'au** *date\_et\_heure\_limite*

- Ligne suivante : les informations concernant les alarmes, s'il y en a, sous la forme :

**Alarmes définies :** *position1*, *position2*, ....

En fin de fichier, afficher le nombre d'événements enregistrés, ainsi que le nombre d'événement de chaque catégorie.

## Exemple

A partir du fichier `exemple_complet` donné précédemment, vous devez obtenir la présentation du fichier `resultat_final.pdf` (aux polices, espacements, tailles de caractères, alignements près).

Remarque : il n'est pas demandé de trier les événements par catégorie. Le fait que tous les événements soient rassemblés par catégorie dans le fichier `resultat_final` vient du fait qu'ils le sont dans le fichier `exemple_complet`.

## Travail à rendre à la fin de votre TP5

Voir les dates sur plubel, dans la section « Planning des CM-TD-TP ».

Fichiers à rendre sur plubel en utilisant le lien « Dépôt étape 5 projet ». Pensez à mettre votre numéro de groupe dans l'un des fichiers. Ajoutez un `makefile` ou un fichier texte dans lequel vous donnerez les commandes de compilation. Regroupez tous vos fichiers (`lex`, `yacc`, etc.) dans une archive zip et rendez un fichier par groupe de projet.