

L3 informatique, 2022-2023, Projet de LFC

Traduction iCal en html

Description générale du projet

Le but du projet est d'analyser un fichier iCal, d'en extraire certaines informations et de créer un fichier html dans lequel les événements sont triés chronologiquement.

1 Modalités

Vous devrez rendre le projet par étapes (travaux à rendre toutes les 1, 2 ou 3 semaines selon les étapes). Un sujet spécifique à chaque étape sera mis en ligne sur plubel avec indication de la date limite pour rendre celle-ci.

Lorsque vous rendez une étape, pensez à mentionner le numéro de groupe de projet dans le document principal.

Pour les étapes où il faut rendre un programme, vous devez fournir les instructions nécessaires à la création de l'exécutable et ne pas oublier de remettre tous les modules nécessaires à la compilation et l'exécution de votre programme (.h, .c, etc.). Inutile de fournir un exécutable ; il ne fonctionnera pas. Si vous fournissez un makefile, celui-ci ne doit pas inclure l'exécution de votre programme sur un de vos fichiers tests ni la suppression des fichiers C intermédiaires.

Après chaque étape (sauf la dernière), il vous sera communiqué un corrigé de celle-ci afin que vous puissiez réaliser l'étape suivante sur une base correcte.

Afin que nous puissions tester vos programmes, merci de faire en sorte que ceux-ci puissent être compilés et exécutés sur les machines des salles d'enseignement.

2 Fichiers à analyser

Les fichiers iCal permettent les échanges de données de calendrier. Vous trouverez une description complète des fichiers iCal dans l'article « iCalendar » de wikipédia (<https://fr.wikipedia.org/wiki/iCalendar>) mais, **pour la réalisation de ce projet, vous devez prendre en compte uniquement la description simplifiée donnée dans cet énoncé.** En particulier, on s'intéressera uniquement aux événements (pas aux tâches, ni aux entrées de journal).

Un fichier iCal commence par **BEGIN:VCALENDAR** et termine par **END:VCALENDAR**.

Chaque description d'événement du calendrier commence par **BEGIN:VEVENT** et termine par **END:VEVENT**.

Entre **BEGIN:VCALENDAR** et la description du premier événement, on peut trouver plusieurs lignes d'informations qui ne sont pas utiles pour le traitement qui vous est demandé. Ces informations devront donc être ignorées par votre analyseur.

Entre le **BEGIN:VEVENT** et le **END:VEVENT** d'un événement, on trouve les informations sur l'événement. Certaines devront être ignorées, d'autres devront être prises en compte pour la création du fichier en html comme, par exemple, le titre de l'événement, la date et heure de début, la date et heure de fin, la description, le lieu, les notifications définies, les répétitions.

Événements uniques (sans répétition), sans alarme, avec heure de début et de fin

Les dates et heures de début et de fin d'un événement unique (sans répétition) sont données respectivement par les lignes :

DTSTART: *date_début* *heure_début* **Z**

et

DTEND: *date_fin* *heure_fin* **Z**

où

date_début et *date_fin* sont des chaînes de caractères de 8 chiffres

heure_début et *heure_fin* sont des chaînes de caractères de 6 chiffres

Le *titre*, le *lieu* et la *description* de l'événement sont des chaînes de caractères quelconques sans retour à la ligne et sans deux-points (:).

Les lignes indiquant ces données sont de la forme :

SUMMARY:*titre*

LOCATION:*lieu*

DESCRIPTION:*description*

Remarque : les informations sur le lieu et la description ne sont pas obligatoires mais s'ils ne sont pas renseignés, on trouve quand même les lignes dans la description de l'événement.

Exemples :

Événement unique commençant à 13h00 le 11/01/2023 et terminant à 14h00 le 11/01/2023 et dont le titre est « RDV dentiste », le lieu est « 3 rue des clématites, Chenôve » et la description est « apporter radios ».	Événement unique commençant à 9h00 le 11/01/2023 et terminant à 11h00 le 11/01/2023 et dont le titre est « cours info », sans lieu ni description.
BEGIN:VEVENT DTSTART:20230111T130000Z DTEND:20230111T140000Z DTSTAMP:20230110T113128Z UID:4lslnlhbeite051mc9j97vivhq@google.com CREATED:20230110T112701Z DESCRIPTION:apporter radios LAST-MODIFIED:20230110T112701Z LOCATION:3 rue des clématites, Chenôve SEQUENCE:0 STATUS:CONFIRMED SUMMARY:RDV dentiste TRANSP:OPAQUE END:VEVENT	BEGIN:VEVENT DTSTART:20230111T090000Z DTEND:20230111T110000Z DTSTAMP:20230110T113128Z UID:36f73v1tne1jghbbbq8ce9uvii@google.com CREATED:20230110T112553Z DESCRIPTION: LAST-MODIFIED:20230110T112553Z LOCATION: SEQUENCE:0 STATUS:CONFIRMED SUMMARY:cours info TRANSP:OPAQUE END:VEVENT

Remarque : Les parties en orange sont à ignorer.

Alarmes

Si une alarme a été programmée pour rappeler un événement, celle-ci est décrite juste avant le **END:VEVENT** entre les mots clés **BEGIN:VALARM** et **END:VALARM**. Dans la description d'une alarme, la seule ligne intéressante est celle de la forme :

TRIGGER:*description_alarme*

Cette ligne indique combien de temps avant le début de l'événement l'alarme doit sonner/s'afficher. Elle est de la forme **-Pn1DTn2Hn3Mn4S** où *n1*, *n2*, *n3*, et *n4* donnent respectivement le nombre de jours, d'heures, de minutes et de secondes avant événement auxquels l'alarme doit se déclencher.

On peut programmer plusieurs alarmes pour un même événement, auquel cas elles seront toutes énumérées au même endroit les unes en dessous des autres.

Exemples :

Événement unique avec 1 alarme programmée 1 heure avant.	Événement unique avec 2 alarmes programmées 30 minutes et 10 minutes avant.
BEGIN:VEVENT DTSTART:20230112T070000Z DTEND:20230112T090000Z DTSTAMP:20230110T113128Z UID:4un01im64v5a04b3gk69m00h4v@google.com CREATED:20230110T112744Z DESCRIPTION: LAST-MODIFIED:20230110T112744Z LOCATION: SEQUENCE:0 STATUS:CONFIRMED SUMMARY:TD info TRANSP:OPAQUE BEGIN:VALARM ACTION:DISPLAY DESCRIPTION:This is an event reminder TRIGGER:-P0DT1H0M0S END:VALARM END:VEVENT	BEGIN:VEVENT DTSTART:20230112T120000Z DTEND:20230112T130000Z DTSTAMP:20230110T113128Z UID:18dtb9vg3gmo3oqp002j1f8pei@google.com CREATED:20230110T112827Z DESCRIPTION: LAST-MODIFIED:20230110T112827Z LOCATION: SEQUENCE:0 STATUS:CONFIRMED SUMMARY:RDV prof TRANSP:OPAQUE BEGIN:VALARM ACTION:DISPLAY DESCRIPTION:This is an event reminder TRIGGER:-P0DT0H10M0S END:VALARM BEGIN:VALARM ACTION:DISPLAY DESCRIPTION:This is an event reminder TRIGGER:-P0DT0H30M0S END:VALARM END:VEVENT

Evénements répétitifs

On peut également enregistrer des événements répétitifs. Lorsqu'un événement est défini avec répétitions, les lignes indiquant le début et la fin de l'événement sont de la forme :

DTSTART;TZID=fuseau_horaire:date_débutTheure_début

et

DTEND;TZID=fuseau_horaire:date_finTheure_fin

au lieu de

DTSTART: date_débutTheure_débutZ

et

DTEND: date_finTheure_finZ

et une ligne supplémentaire indiquant la répétition choisie se trouve dans la description de l'événement sous la forme :

1) RRULE:FREQ=frequence;WKST=SU;COUNT=nb;BYDAY=liste_jours_semaine

ou

2) RRULE:FREQ=frequence;UNTIL=date_limiteTheure_limiteZ

selon que l'on indique le nombre de répétitions ou la date jusqu'à laquelle on veut répéter.

Avec :

frequence : une chaîne de caractères parmi {DAILY, WEEKLY, MONTHLY, YEARLY} ;

nb : un nombre entier ;

liste_jours_semaine : une suite de chaînes de caractères prises parmi {SU, MO, TU, WE, TH, FR, SA} et séparées par des virgules.

date_limite : date au même format que *date_debut* et *date_fin*

heure_limite : heure au même format que *heure_debut* et *heure_fin*

Explications :

FREQ=WEEKLY indique que l'événement se répète toutes les semaines

WKST=SU indique que le changement de semaine se fait le dimanche (SU = Sunday). C'est une simplification car, selon les pays, le changement de semaine ne se fait pas le même jour. Pour le projet, on supposera qu'on a toujours SU.

COUNT=8 indique que l'événement se répète 8 fois

BYDAY=MO, WE, FR indique que l'événement se répète les lundis (MO), mercredi (WE) et vendredi (FR)

UNTIL=20230502T100000Z indique que l'événement se répète jusqu'au 2 mai 2023, 10h.

Remarque : on peut aussi trouver les options WKST et BYDAY dans une règle de la forme 2).

Exemples :

Evénement répété 3 fois, tous les vendredis de 8h à 10h à partir du 13/01/2023.	Evénement répété tous les jours de 14h à 16h entre le 16 et le 20 janvier 2023 (inclus).
BEGIN:VEVENT DTSTART;TZID=Europe/Paris:20230113T080000 DTEND;TZID=Europe/Paris:20230113T100000 RRULE:FREQ=WEEKLY;WKST=SU;COUNT=3;BYDAY=FR DTSTAMP:20230110T113128Z UID:242gpg7ejoh5m13tenpuekblb5@google.com CREATED:20230110T112928Z DESCRIPTION: LAST-MODIFIED:20230110T112928Z LOCATION: SEQUENCE:0 STATUS:CONFIRMED SUMMARY:TP info TRANSP:OPAQUE END:VEVENT	BEGIN:VEVENT DTSTART;TZID=Europe/Paris:20230116T140000 DTEND;TZID=Europe/Paris:20230116T160000 RRULE:FREQ=DAILY;UNTIL=20230220T225959Z DTSTAMP:20230110T113128Z UID:7va6mfe0m39pm77ho0tfqhk3sc@google.com CREATED:20230110T113037Z DESCRIPTION: LAST-MODIFIED:20230110T113037Z LOCATION: SEQUENCE:0 STATUS:CONFIRMED SUMMARY:TD gestion TRANSP:OPAQUE END:VEVENT

Evénements sur une journée

Les événements sur une journée complète ne comportent pas d'heure de début ni de fin. Les lignes indiquant le début et la fin de l'événement sont de la forme :

DTSTART;VALUE=DATE:date_début

et

DTEND;VALUE=DATE:date_fin

au lieu de

DTSTART: date_débutTheure_débutZ

et

DTEND:date_finTheure_finZ

Exemple :

Evénement sur toute la journée du 14 janvier 2023. Remarque : la date de fin est le lendemain.

BEGIN:VEVENT

DTSTART;VALUE=DATE:20230114

DTEND;VALUE=DATE:20230115

DTSTAMP:20230110T113128Z

UID:12j9po9pcke9reiec7l88dhjoc@google.com

CREATED:20230110T113114Z

DESCRIPTION:

LAST-MODIFIED:20230110T113114Z

LOCATION:

SEQUENCE:0

STATUS:CONFIRMED

SUMMARY:journée portes ouvertes

TRANSP:TRANSPARENT

END:VEVENT

3 Etapes du projet

Analyse lexicale

La première étape consistera à établir la liste des unités lexicales du langage et donner la description de celles-ci.

Cette liste sera utilisée lors de la seconde étape pour l'écriture du programme lex qui permettra de générer l'analyseur lexical de votre compilateur.

Analyse syntaxique

Les étapes suivantes consisteront à décrire de façon formelle la syntaxe d'un fichier iCal et écrire le programme yacc à partir duquel sera généré l'analyseur syntaxique du compilateur.

Analyse sémantique/traduction

Les dernières étapes consisteront à insérer dans les programmes lex et yacc les instructions permettant de traduire le fichier iCal en html. La forme du fichier html à restituer vous sera donnée à ce moment-là.

Table des symboles

La construction de la table des symboles commencera après l'écriture du programme lex. Dans cette structure seront stockées toutes les informations sur les différents événements trouvés dans le fichier source.