

Génération outillée de planning Latex attribution de grammaire

Université Lille1
UFR IEEA

Licence S5
TP COMPIL – 2011-2012

FIL

Objectif Ce but de ce TP est d'attribuer la grammaire réalisée au TP précédent pour générer la version Latex des plannings.

Matériel fourni Récupérer sur le portail l'archive `tp4.tgz` qui contient un répertoire principal `init` donné à titre d'exemple et un répertoire principal `planning`.

1 Matériel fourni : l'exemple d'Init

L'exemple fourni est le contrôle de type de INIT présenté sur les transparents.

Plusieurs fichiers `.cup` sont fournis dans `tp4.tgz` :

- `anSyntInit.cup` : la grammaire attribuée, qualifiée de peu lisible
- `anSyntInit_sansActions.cup` : la grammaire non attribuée (pour relire facilement la syntaxe)
- `anSyntInit_plusPropre` : la grammaire attribuée de manière plus lisible, en passant par une classe Java dont la responsabilité est le contrôle de type

L'analyse sémantique se lance en lançant l'analyseur syntaxique.

2 Travail à réaliser

Ne pas oublier le source `envm5.sh`. Créer un nouveau répertoire pour ce TP, copie du répertoire du TP précédent, puis ajouter les nouveaux fichiers `.cup` pour INIT.

2.1 Découverte de Cup avec Init

Depuis le répertoire `init` :

- regarder la spécification `anSyntInit.cup` pour apprendre la syntaxe de CUP pour les grammaires attribuées (appelée dans les transparents) ;
- supprimer momentanément la déclaration d'un attribut pour constater que l'erreur apparaît au moment de compiler ou d'exécuter le code généré par CUP.

2.2 Génération Latex pour les plannings

Depuis le répertoire principal `planning` :

- recopier dans `spec` les spécifications du TP précédent pour l'analyse lexicale et syntaxique ;
- attribuer la grammaire que contient le fichier `.cup` jusqu'à générer complètement les fichiers latex ;
- à chaque modification régénérer l'analyseur syntaxique par un `genererAnalyseurSyntaxique.sh` et compiler.

Comme cette génération de Latex ne comporte aucune vérification sémantique, les tests négatifs n'ont pas de sens ici.

Vous devez obtenir une grammaire attribuée la plus lisible possible. Une approche possible est de commencer par écrire une attribution peu lisible puis, une fois que les tests sont probants, de réusiner la grammaire sur le modèle d'INIT.

À rendre sur PROF N'oubliez pas d'écrire les noms du binôme dans les fichiers `.lex` et `.cup`.

Rendre une archive de votre projet, incluant les tests, et un `README` expliquant l'avancement de votre travail, ce qui ne marche pas le cas échéant, et répondant aux questions suivantes.

Supposons qu'on veuille effectuer des vérifications sur les plannings avant de générer une affiche contenant des informations bizarres :

1. quelles vérifications proposez-vous ?
2. avez-vous mis en œuvre tout ou partie de ces vérifications lors de l'analyse lexicale ?
3. comment vous y prendriez-vous ?