

HMIN306 : Évolution et restructuration de logiciels
Analyse statique et dynamique

1) Exercice 1 : Découverte de l'AST (JDT)

1. Tutoriel AST : https://www.eclipse.org/articles/article.php?file=Article-JavaCodeManipulation_AST/index.html
2. Parseur AST dans JDT : <http://www.programcreek.com/2011/01/a-complete-standalone-example-of-astparser/>
3. AST View : <http://www.eclipse.org/jdt/ui/astview/index.php>
4. Patron visiteur : [https://fr.wikipedia.org/wiki/Visiteur_\(patron_de_conception\)](https://fr.wikipedia.org/wiki/Visiteur_(patron_de_conception))

2) Exercice 2 : Extraire les informations suivantes à partir de l'AST

1. La liste des classes déclarées dans le code analysé
 1. nom de la classe
 2. liste des attributs
 1. nom de l'attribut
 2. encapsulation de l'attribut : public, protected, private
 3. liste des méthodes
 1. nom de la méthode
2. La liste des classes déclarées dans le code analysé
 1. nom de la classe
 2. nom de toutes les supers classes

3) Exercice 3 : Extraire les informations suivantes du code analysé

1. Pour une classe donnée
 1. la liste de ses méthodes
 1. nom de la méthode
 2. pour chaque méthode
 1. La liste des méthodes appelées par cette méthode
 1. nom de la méthode
 2. Type de l'objet receveur de l'appel

4) Exercice 4 : Construction d'un graphe d'appel

En s'appuyant sur les résultats de l'exercice précédent, construisez le graphe d'appel qui correspond au code analysé

5) Exercice 5 : Utiliser la technique RTA pour construire un graphe d'appel plus précis

6) Exercice 6 : Construire le flot de contrôle d'une méthode donnée

7) Exercice 7 : analyse dynamique. Instrumenter le code pour :

1. exécuter quelques scénarios et afficher les traces d'exécution. Une trace est une liste d'appels
2. construire le graphe d'appel dynamique