

2 minutes – Construction des compilateurs

Etapes du front-end

- 1- Scanner
- 2- Parser + construction de l'AST
- 3- Liaison des noms
- 4- Typage
- 5- traduction en une représentation intermédiaire

Ajout d'une fonctionnalité « const » dans un langage

Spécification lexical : ajout d'un nouveau mot clé « const »

Scanner : rajout de la ligne 'const' { return token ::CONST ;}

Spécification syntaxique : augmenter ou rajout d'une règle

Syntaxe abstraite : ajout d'un attribut booléen

Parser : ajout d'un token et ajout d'une nouvelle production

Liaison des noms : Rien à modifier

Sucre syntaxique

Construction syntaxique supplémentaire proposé par un langage n'apportant rien de plus au niveau de la syntaxe abstraite mais permettant des variantes d'écriture.

Question C++ : Pourquoi les références en C++ peuvent être considérés comme des sucres syntaxiques ?

Une référence est un raccourci pour un pointeur.

Avantage : concision (au niveau de l'opérateur de dé-référencement)

Sureté (obligation d'initialisation, pas d'arithmétique des pointeurs)

Désucre

Plus il s'effectue tôt, moins il y a d'impact sur le compilateur mais le message à l'utilisateur ne sont pas clairs.

Plus il s'effectue tard, plus le sucre est complexe.

C++

La surcharge des fonctions dépend des types à l'exécution : FAUX

L'utilisation des fonctions virtuelles est incompatible avec la compilation séparée : FAUX

Les pré-requis forts pour le typage statique dans un langage orienté objet :

- Liaison des noms résolus à la compilation
- Absence d'instruction de transtypage

Méthode : fonction attachée à un objet, faisant parti de l'interface de cette dernière

Méthode polymorphe : méthode redéfinissant dans les sous-classes de son site de définition, et dont la liaison s'effectue à l'exécution, selon le type dynamique de son objet-cible

Multiméthode : méthode dont la liaison (dynamique) peut dépendre du type dynamique de plus d'un de ses arguments

Simuler le mécanisme d'une multiméthode : utilisation du design pattern visitor

Fonction objet en C++ : Objet qui contient un opérateur ()

Passage d'argument

C : passage par valeur/copie

C++ : par valeur/copie + référence

Tiger : par valeur/copie + référence (pour les tableaux, objets...)

Simuler une référence en Tiger : utiliser des tableaux, objets...

Banque de questions :

- Un compilateur large fait passer toutes les étapes au programme entier les unes après les autres
- Lex/Flex sont des générateurs de scanners
- Yacc/Bison sont des générateurs de parseurs
- Comment désambigüiser exp : exp '+' exp | exp '-' exp | NUM -> %left '+' '-'
- Comment désambigüiser exp : exp '+' exp | exp '*' exp | NUM -> %left '+' %left '*'
- Le métier d'un parseur est de faire l'analyse syntaxique
- Si une grammaire hors contexte est non ambiguë, elle n'est pas nécessairement LL
- Les « start condition » de lex/flex (%s et %x) permettent de supporter différents contextes lexicaux
- Yacc repose sur l'algorithme LALR(1)
- La syntaxe concrète est l'interface homme-machine d'un langage de programmation
- AST : abstract syntax tree – arbre de syntaxe abstraite
- ASN.1 est une syntaxe pour décrire des paquets de données structurées
- Désucre signifie traduire certaines phrases dans une forme plus primitive
- La classe Symbol qui gère les identificateurs permet de manipuler aussi efficacement que s'il s'agissait d'entiers