

SÉANCE 7

INTRODUCTION AUX COMPILATEURS

SUJETS

Langages naturels

- Entités lexicales
- Syntax et sémantiques

Langages d'ordinateur

- Analyse lexicale
- Analyse syntaxique
- Analyse sémantique

Compilateurs

- Exigences de base du compilateur
- Processus de compilation

LANGAGES NATURELS - BASE

Dans un langage (naturel):

- Une phrase est une séquence de mots
- Un mot (aussi appelé unité lexicale) est une séquence de caractères (possiblement un seul)

L'ensemble de caractères utilisés dans un langage est fini (connu sous le nom d'alphabet)

L'ensemble de phrases possibles dans un langage est infini

Un dictionnaire liste tous les mots (entités lexicales) d'un langage

- Les mots sont classifiés en différentes catégories lexicales: verbe, nom, pronom, préposition....

LANGAGES NATURELS - BASE

La grammaire (aussi considérée l'ensemble de règles syntaxiques) est utilisée pour déterminer quelles séquences des mots sont bien formées

- Les séquences doivent avoir une structure qui obéit au règles de grammaire

Les phrases bien formées ont généralement un sens que les humains comprennent

On essaie d'apprendre aux machines notre langage naturel



***Avec des
résultats
mixtes!!***

ANALYSE DES PHRASES

Analyse lexicale: identification de mots formés de caractères

- Les mots sont classifiés en plusieurs catégories: articles, noms, verbes, adjectifs, prépositions, pronoms...

Analyse syntaxique: règles pour combiner des mots afin de former des phrases

Analyse du sens: difficile à formaliser

- Facile pour les humains
- Difficile pour les machines (malgré l'évolution du traitement de langage naturel)
 - Grand domaine de recherche pour ceux intéressés dans les études supérieures...

TRAITEMENT DU LANGAGE D'ORDINATEUR

Dans les langages d'ordinateur (ou programmation), on parle d'un programme (correspondant à une phrase longue ou paragraphe)

- Séquence d'unités lexicales
- Les unités lexicales sont des séquences de caractères

Les règles lexicales d'un langage déterminent quelles sont les unités lexicales valides de ce langage

- Il existe plusieurs **catégories lexicales**: identificateur, nombre, chaînes de caractères, opérateur...
- Les catégories lexicales sont aussi nommées « **tokens** »

TRAITEMENT DU LANGAGE D'ORDINATEUR

Les règles syntaxiques d'un langage déterminent quelles séquences d'unités lexicales sont des programmes bien formés

Le sens d'un programme bien formé est aussi appelé sémantiques

- Un programme peut être bien formé, mais ses énoncés ne font pas de sens
- Exemple:

```
int x = 0;  
x = 1;  
x = 0;
```
- Syntaxiquement, le code ci-haut est valide, mais quel est son sens??

TRAITEMENT DU LANGAGE D'ORDINATEUR

Les compilateurs doivent attraper les erreurs lexicales et syntaxique

Les compilateurs peuvent se plaindre des erreurs sémantiques communes:

```
public boolean test (int x){  
    boolean result;  
    if (x > 100)  
        result = true;  
    return result;  
}
```

 **Message d'erreur:**
The local variable result may
have not been initialized

Vos collègues ou bien le client se plaindront du reste!!

COMPILATEURS

Définition d'un compilateur

- Un programme qui traduit un programme exécutable dans un langage vers un programme exécutable dans un autre langage
- On prévoit que le programme produit par le compilateur soit meilleur que l'original

Définition d'un interpréteur

- Un programme qui lit un programme exécutable et produit les résultats qui proviennent lorsqu'on exécute ce programme

**On se concentre sur les compilateurs dans ce cours
(toutefois, beaucoup de concepts s'appliquent aux deux)**

EXIGENCES DE BASE POUR LES COMPILATEURS

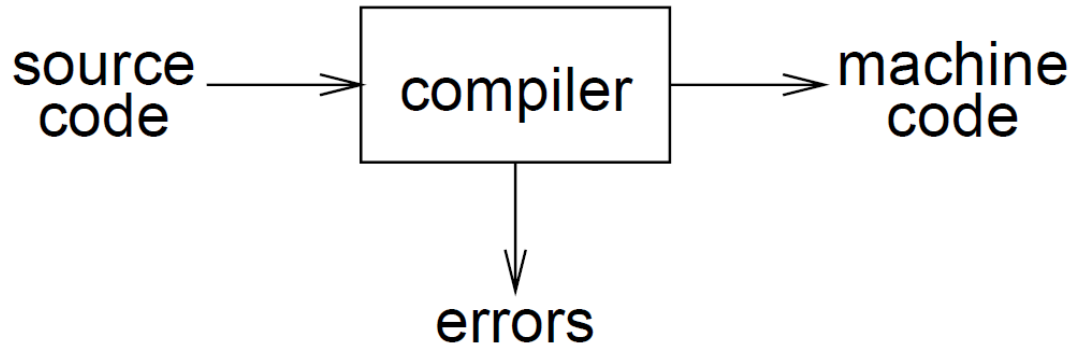
Le compilateur doit faire:

- Produire le code correct (code byte dans le cas de Java)
- Exécuter rapidement
- Le code produit doit être optimale
- Achever un temps de compilation proportionnel à la taille du programme
- Bien travailler avec les débogueurs (*absolument nécessaire*)

Le compilateur doit avoir:

- Bonne diagnostique pour les erreurs lexicales et syntaxiques
- Support pour appel dans autres langages (*voir Java Native Interface si vous êtes intéressés*)

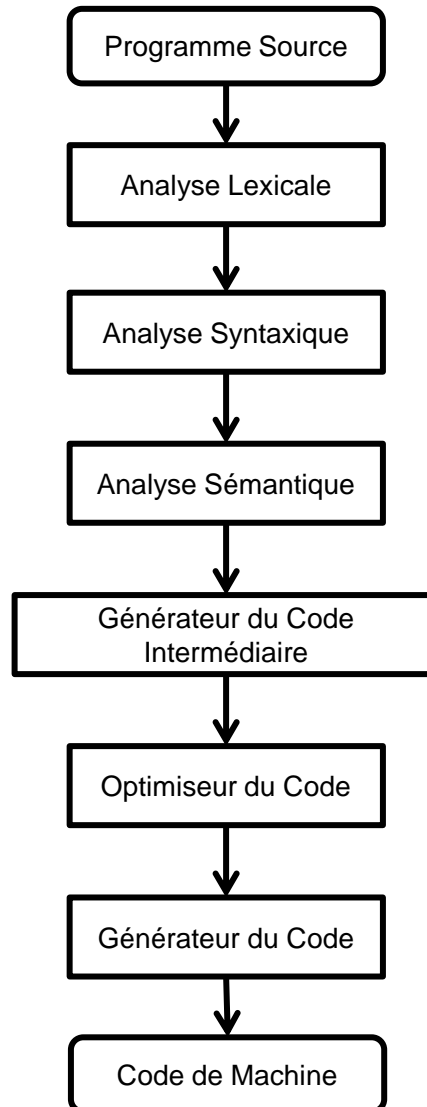
VUE ABSTRAITE DES COMPILATEURS



Un compilateur réalise une traduction en plusieurs étapes; par correspondance, il contient plusieurs composantes

Généralement, un compilateur inclut (au moins) des composantes séparées pour vérifier les règles lexicales et syntaxiques

PROCESSUS DE COMPILATION



PROCESSUS DE COMPILE

Un cours complet est requis afin de couvrir les détails des phases différentes

Dans ce cours, nous voyons seulement une introduction

- On se concentre sur l'analyse lexicale et syntaxique

QUELQUES DÉFINITIONS IMPORTANTES

Ces définitions, malgré qu'elles sont ennuyantes, sont importantes afin de comprendre les concepts qui seront introduits dans les prochaines séances

Commençons...

ALPHABET

Souvenez-vous du début de la séance (ou de la maternelle):
l'alphabet est un ensemble de caractères qui sont utilisés
pour former un mot



Puisque les mathématiciens adorent les symboles Grecs, on
va référer à l'alphabet avec Σ

ALPHABET

Σ est un alphabet, ou un ensemble de caractères

- Ensemble fini et consiste de tous les caractères d'entrée ou symboles qui peuvent être arrangés pour former des phrases dans un langage

Langage de programmation: généralement un ensemble d'ordinateur bien-défini comme le ASCII

CHAÎNES DANS UN ALPHABET

$\Sigma = \{a, b, c, d\}$

Chaînes possibles de Σ incluent:

- aaa
- aabbccdd
- d
- cba
- abab
- cccccccccacccc
- Malgré que ceci est amusant, je pense que vous saisissez l'idée...

LANGAGES FORMELS

Σ : alphabet, c'est un ensemble fini composé de tous les caractères d'entrée ou symboles

Σ^* : fermeture de l'alphabet, l'ensemble de toutes les chaînes possibles dans Σ , incluant la chaîne vide ε

Un langage (formel) est un sous-ensemble spécifique de Σ^*

MERCI!

QUESTIONS?