

Histoire de l'informatique

Histoire des langages

Pré-langages.:

- Exemples : Langage machine, assembleur.
- Avantage : ils sont proches de la machine.
- Inconvénients : ils sont peu fiables.

Les ancêtres, dinosaures :

- Fortran nov. 1954, Cobol 1959 : compilé, non structurés
- Basic mai 1964 : interprété.

Les structurés :

- Lisp 1958 : interprété, structuré.
- C 1971 (faiblement typé).
- Pascal 1970 (typé)

Langages objets :

Smalltalk 1969, C++ jui. 1983, Borland Pascal objet 1988, Basic.

Environnements de développement intégrés pour Windows :

Visual C++, Visual Basic, Borland, WinDev.

Environnements de développement intégrés pour le Web :

.net Java,

Histoire des systèmes d'exploitation

Premiers ordinateurs

Conrad Suze ?

Allemand
Sur une table de cuisine
Mécanique ?

Mark1 Howard Aiken

Au début de la guerre
Ordinateur à relais
100 ms de temps de cycle

ENIAC (Electronical & Numeral Integrater And Computer)

Prosper Eckert, John Mauchly.
A lampes (bugs=Papillon), chauffe, rempli une salle.
Temps de cycle : 1ms.
Calculs balistiques, à la fin de la guerre.

EDVAC (Electronical Discrete Variable Computer)

Après la guerre.
Structure John von Neumann.

Mémoires câblées

L'Unité Arithmétique et Logique (avec ses Registres) faisait la taille d'une table.
Mémoire de masse à tambours magnétiques.
Pas encore de mémoires à tores de ferrite.
Programmation par câblage du traitement à la main, (JFL : au fer à souder).

Bootstrap

Rentrer le premier bootstrap aux clés.
Pour lire un second bootstrap.

Batch

Traitement des fichiers par lots (cartes ou bandes perforées)
Bandes magnétiques en entrée.
Grosses imprimantes en sortie (EBR ou LBR de 3M)

Unix 1970

Le langage C a été inventé par Dennis Ritchie pour le système d'exploitation UNIX (Laboratoire BELL).

Exociser pour le 6800 (1978 ?)

CPM

Pour le Zilog 80

IBM PC 1980 1981 ?

DOS Bill Gates 1980 1981 ?

Disk Operating System, et les premiers piratages.

Apple, Lisa Mac 1984

Dans un garage.
Steve Wozniak (28 ans ?) Le technicien.
Steve Job, (18 ans) Le commercial.

Mac 1984

PARC, Palo Alto Research Center, Alan Kay.
Graphisme, et souris (Edelman ?).

Windows 1985

Bill Gates, espionnage industriel.

Mini Unix

Linux 1995 ?

Linus Torvald
Unix, multi-tâches.

Smalltalk, Squeak

Alan Kay, lisp, scheme
Interpréteur

Java

Machine virtuelle

Net

HTML, PHP.

Définissons quelques notions

Langage structuré

- Différence entre la structure d'arbre et celle de réseau.
- Paradoxe : diminuer la liberté pour augmenter la fiabilité.
- Fiabilité : car le programme est parcourable systématiquement.

Langage fonctionnel.

- *Privilégier les fonctions qui créent un contexte local plutôt que global.*

Contexte local :

Exemple d'une fonction : marche(23) ;

Variables locales dans le temps et dans l'espace.

- Évaluation de $((1+2) * (6/3))$



- Les fonctions agissent par remontée de valeur plutôt que par effet de bords.

Contexte global :

- Pour comprendre la notion de contexte global, citer des exemples en français, en programmation, en HTML.
- Les fonctions agissent par effet de bords, plutôt que par remontée de valeur

- *Paradoxe : diminuer la liberté du programmeur augmenter la fiabilité du logiciel !*

Quand il n'y a pas d'effets de bord, la fiabilité est augmentée car tout est bien en vue dans le contexte local, il n'y a pas d'effets à distance, qu'on finit par perdre de vue.

Implémentation

L'implémentation de telles structure pour le passage de paramètres se fait dans une pile

Typage des langages fonctionnels.

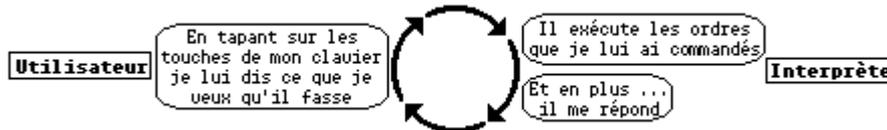
- Le typage introduit une sémantique qui permet de trouver des erreurs dans les programmes.
- Le paradoxe du typage : diminuer la liberté pour augmenter la fiabilité.
- Les erreurs apparaissent dès la compilation. Le temps que l'on perd à compiler est largement récupéré car le débogage à l'exécution est plus rapide.

Une structure de pile pour implémenter un langage structuré compilable

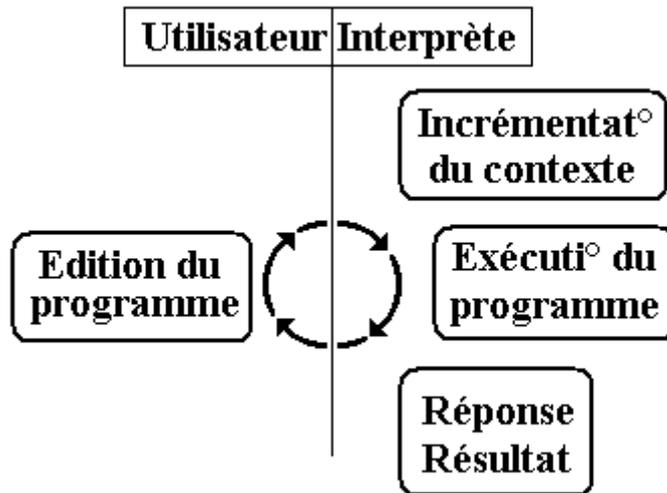
Différence entre langage interprété et compilé

Boucle d'interaction du langage interprété :

Interprète en mode direct



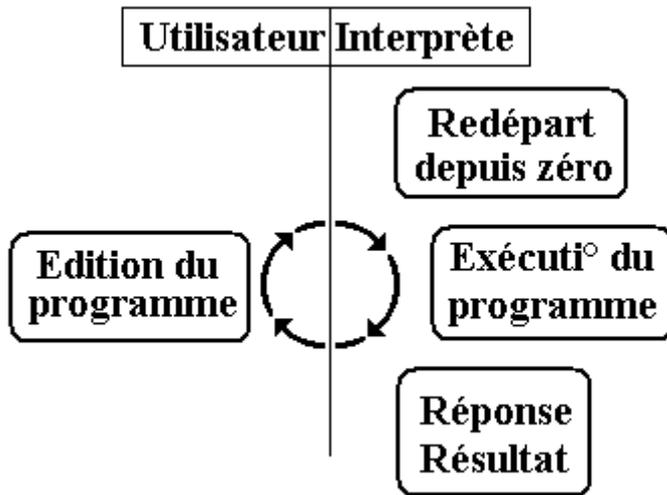
Imprime la valeur de la lecture.



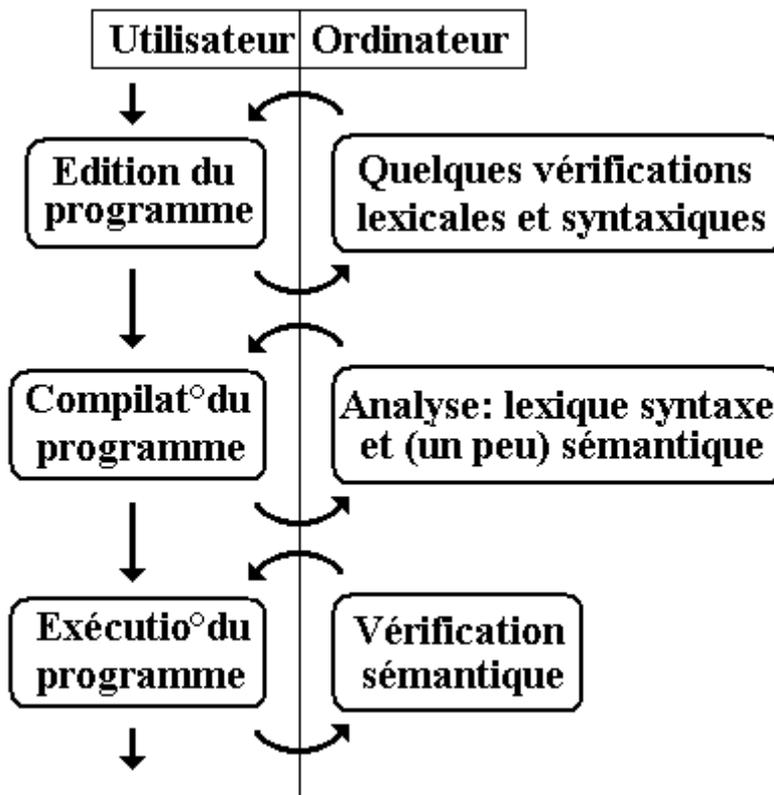
Lecture-analyse lexicale et syntaxique, évaluation, impression. Il fonctionne en mode incrémental.

Interprète en mode éditeur

Il fonctionne en mode hors contexte. Le contexte est RAZéro avant l'exécution du programme.



Complexité du langage compilé :



Edition
 Compilation
 Exécution

Notion de compilation au travers de la notation polonaise inverse
 D'une pierre 3 coups :

- Evaluation interprété de $((1+2) * (6/3))$
 . Elle se fait par le parcours d'un arbre infixé pour interpréter.
- Parcours d'un arbre infixé pour compiler et générer un code postfixé.

Exemple l'implémentation au moyen d'une pile.