



# Les langages de programmation

Min'formatique



# Une définition

Faire des choses :

- Processeur
- Système d'exploitation
- Navigateur internet

Langage de programmation

Avoir une syntaxe :

- Langue parlée
- Expressions mathématiques

# Fonctionnement

Principe  
général :

Langage de  
programmation

Représentation  
intermédiaire

Acteur

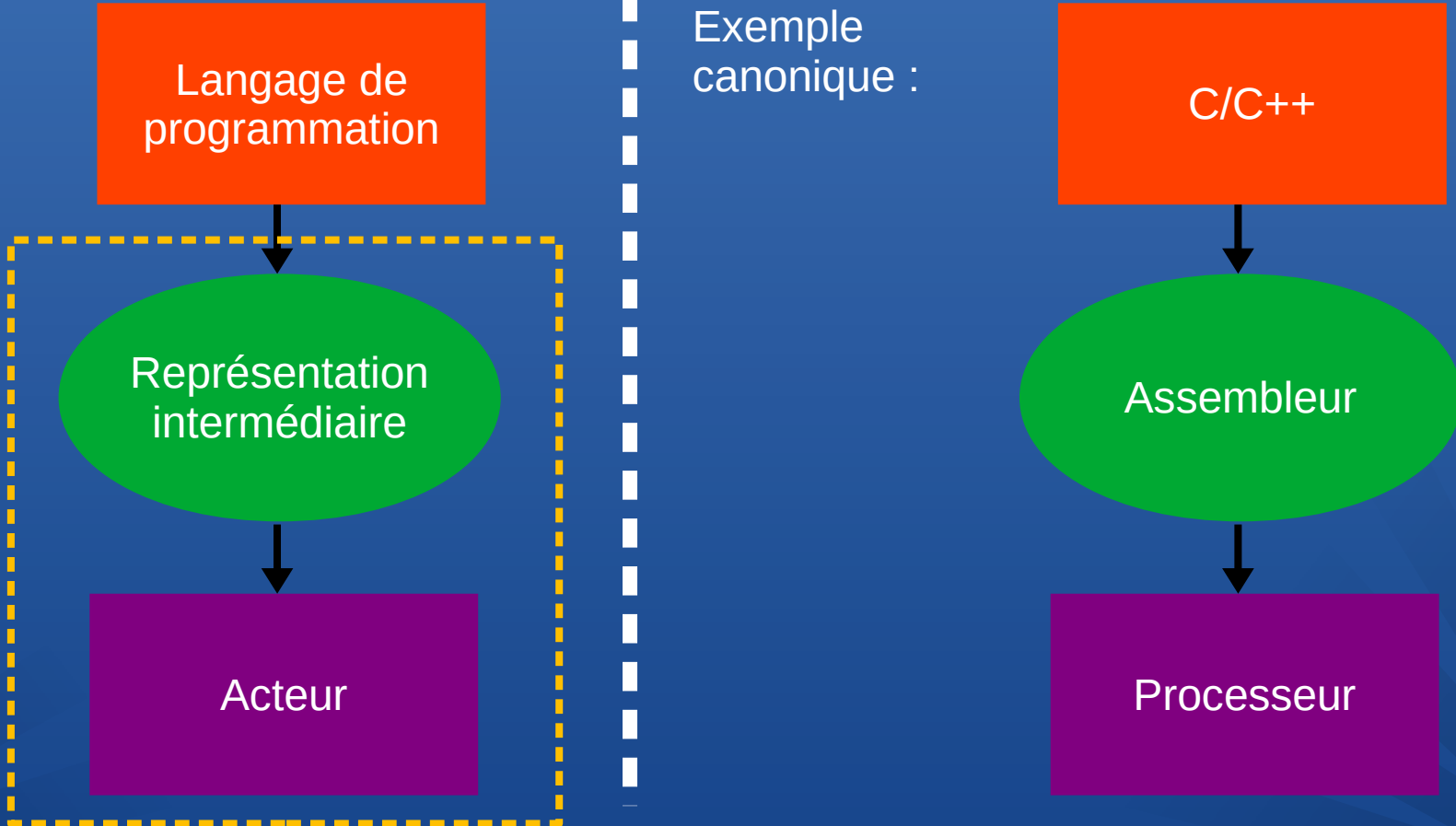
Souvent  
faits par la  
même  
personne

Exemple  
canonique :

C/C++

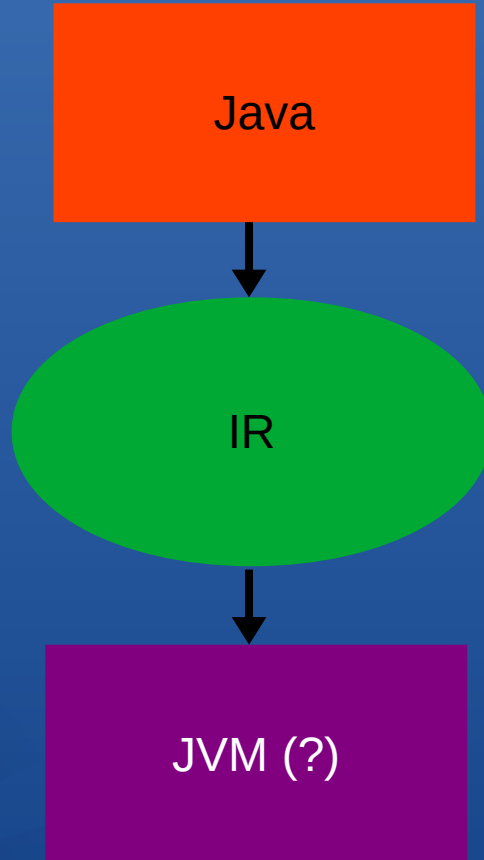
Assembleur

Processeur

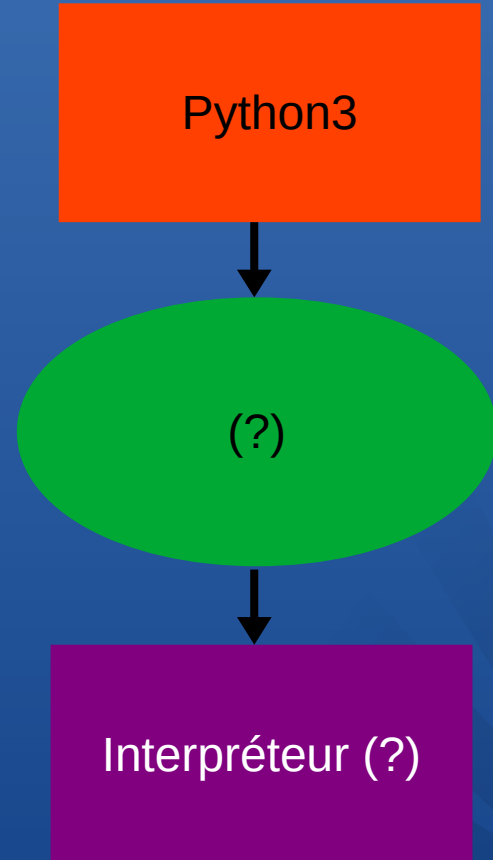


# Exemples plus difficiles

Correct  
techniquement,  
mais Java n'a pas  
été créé pour  
automatiser la  
JVM.



Python est un  
langage de  
script, donc il  
n'a (souvent)  
pas de  
représentation  
intermédiaire.



# Créer un langage

Etape 1	Etape 2	Etape 3	Etape 4
Trouver une raison	Trouver un représentation intermédiaire (si besoin)	Définir une syntaxe	Implémenter
“Ecrire du code en assembleur est pénible”	L’assembleur	K&R C, ANSI C, ISO C, C11	Compilateur C

# Pourquoi créer un langage ?

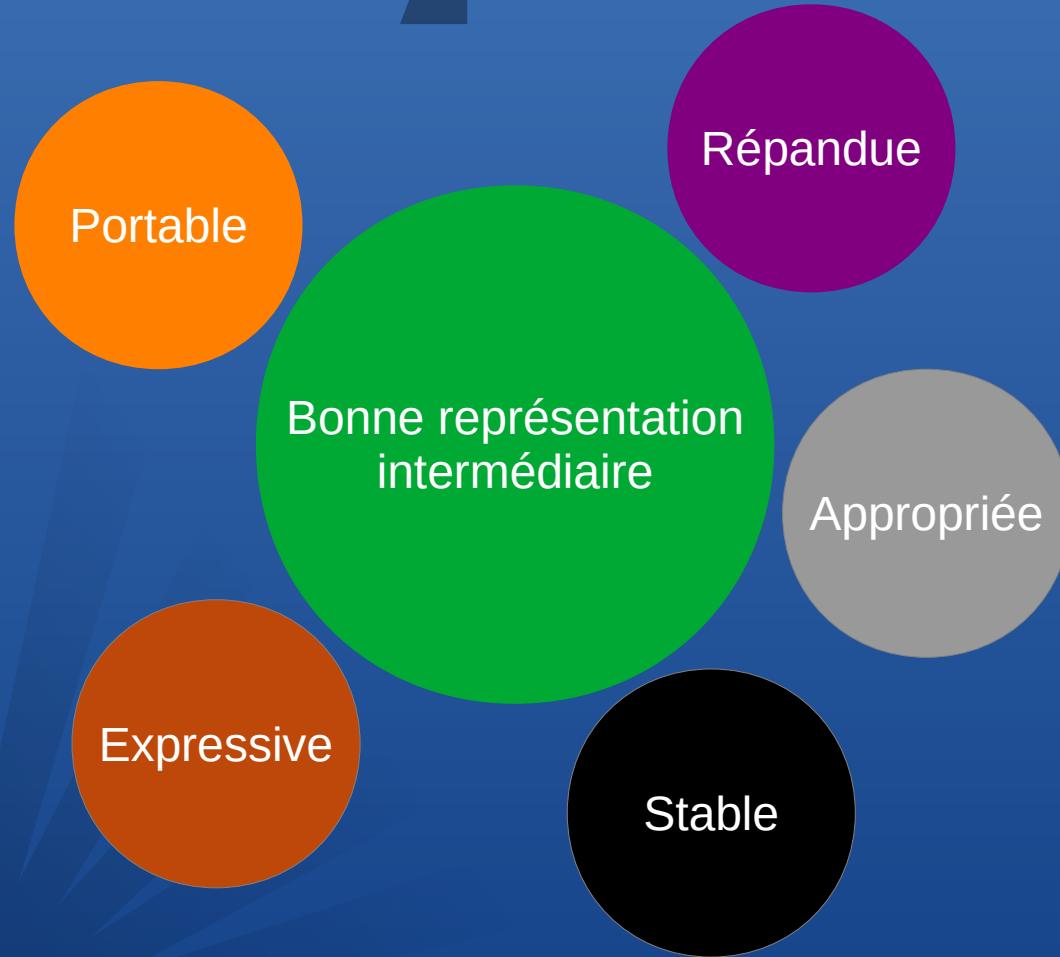
Automatiser  
Un type de tâche  
Effectué par la machine  
ou un humain:  
(C, shell etc...)

Décrire un objet complexe:  
(LateX, Verilog, Esterel etc...)

Pour le challenge:  
(Brainfuck, Ascii dots etc...)

Pour faire mieux qu'un  
Langage existant :  
(C++, Rust, Haskell,  
Java etc...)

# Choisir une représentation



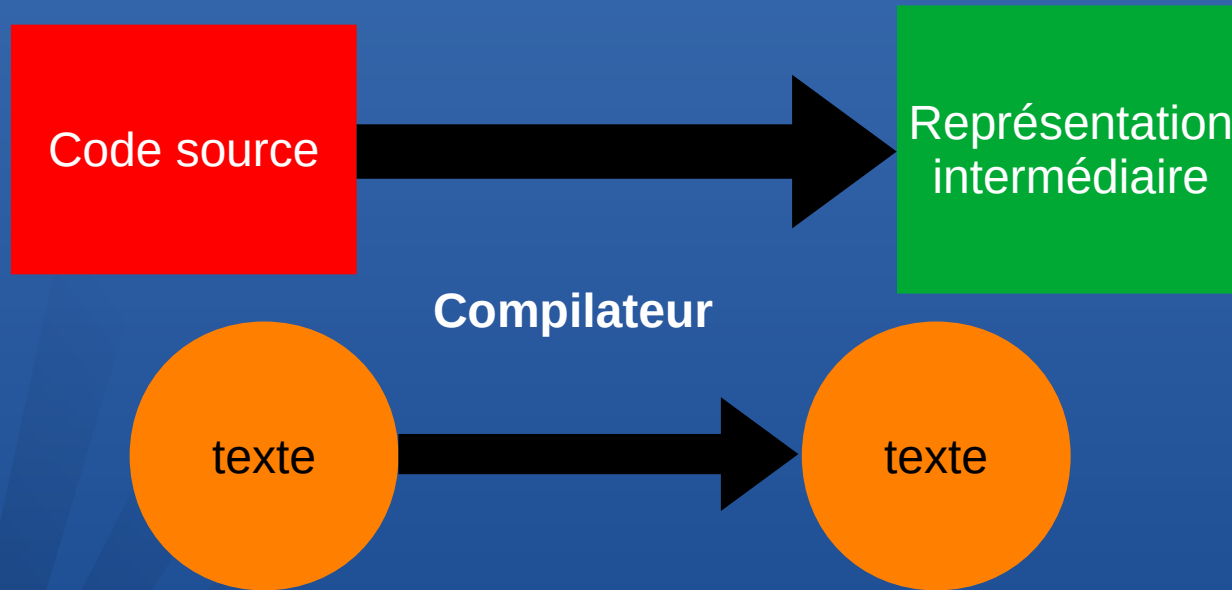
Exemple de **bonne** représentation intermédiaire :

- L'assembleur
- LLVM-IR (probablement ce dont vous avez besoin)
- JVM-IR
- WASM

Exemple de représentation intermédiaire **discutable** :

- Le C
- Verilog
- Javascript

# Comment l'implémenter ?



2 composants d'un compilateur :

- Le lexer transforme le texte en une suite de token identifiés.
- Le parser relie les token à la syntaxe correspondante (i.e. il les interprète dans leur contexte).





# Les langages à connaître

# C

- Ecriture en 1972 par K&R, fort impact historique.
- Lié au système d'exploitation UNIX.
- Bas-niveau, sert à écrire automatiquement de l'assembleur.
- Langage phare de l'Open Source et des systèmes embarqués.

# C++

- Apparition en 1985, extension orienté objet du C.
- Assez critiqué pour son aspect fouillis, a apporté beaucoup d'innovations bienvenues sur le C.
- Langage qui a beaucoup perdu de son intérêt aujourd'hui, il est surtout utilisé pour des raisons historiques.

# Java

- Introduit en 1995 par Sun Microsystems (devenu Oracle), langage orienté objet par excellence.
- Utilisé pour l'écriture d'applications assez grandes et portables vers plusieurs cibles.
- Le langage a été très dominant à son époque mais est aujourd'hui assez impopulaire, en grande partie à cause des mauvaises pratiques commerciales d'Oracle à une époque.

# Python

- Première apparition en 1991, python3 date de 2008.
- Langage de script interprété originaire de Linux avec une syntaxe très riche et orienté objet proche du pseudo code.
- Très utilisé par la communauté scientifique notamment dans le domaine du machine learning, il est également populaire dans l'Open-Source.

# Javascript

- Apparition en 1995 avec le navigateur Netscape (ancêtre de Firefox).
- Prototype écrit en 10 jours, le langage est l'un des piliers du Web. Originellement utilisé comme code client (exécuté sur un navigateur), des versions serveurs en sont rapidement apparues, dont la plus influente node.js en 2009.
- Javascript est aujourd'hui le langage de programmation le plus utilisé selon plusieurs métriques (GitHub en particulier).
- Javascript est la représentation intermédiaire du Web, mais le langage est férocelement critiqué et beaucoup souhaitent le remplacer par une alternative plus solide (WASM notamment).



# Langages de calcul De haute performance

# Fortran

- Le plus ancien langage de programmation continuellement en service, Fortran est apparu en 1956 chez IBM.
- Le langage est dédié au calcul scientifique de haute performance, et on le trouve encore en usage dans les supercalculateurs pour sa robustesse et ses très bonnes performances.
- Les bibliothèques mathématiques de référence BLAS et LAPACK ont été écrites en Fortran à l'origine. L'intérêt du langage est surtout historique et culturel.



# Julia

- Langage moderne apparu en 2012 dédié au calcul de haute performance.
- Le langage a pour but de supplanter le C et Fortran pour le calcul de haute performance.
- Julia est avec le C, le C++ et Fortran le 4ème langage dans lequel des calculs de plus de 1 petaFlops ont été effectués.



# Langages pour systèmes embarqués

# Rust

- Développé en 2010 par les ingénieurs de Mozilla, Rust est l'un des langages les plus aimés dans le monde de l'Open-Source.
- Le Rust est un concurrent direct du C++ pour toutes les applications qui sont trop grosses pour être écrites en C mais incompatibles avec Java.
- La particularité du Rust est d'être Safe, c'est à dire d'éliminer toute une classe de bugs dès la compilation. Cela permet notamment de raccourcir considérablement le temps de débogage par rapport à du code en C/C++.

# Verilog

- Le Verilog est un langage apparu en 1984 qui occupe une place à part dans les langages de programmation en tant qu'HDL (Hardware Description Language).
- Le Verilog est utile pour faire du digital design, c'est à dire notamment concevoir des processeurs. Le code d'un fichier Verilog sert à décrire la logique d'un circuit électronique et même générer le circuit correspondant.
- Le Verilog est aussi utilisé pour programmer des FPGA, qui vont servir soit à développer des ASIC ou processeurs, soit (pour des applications de faible volume à forte valeur ajoutée) pour être utilisés dans des systèmes embarqués.
- Le Verilog est souvent utilisé comme représentation intermédiaire en design digital.



# Langages inattendus mais utiles

# LateX

- LateX est une extension du langage de programmation TeX écrit par Donald Knuth en 1978.
- LateX sert à faire automatiquement de beaux documents à partir de code et est notamment utilisé pour générer des PDF et réputé pour permettre de décrire de belles formules.
- Il n'y a que très peu de compilateurs disponibles pour ce langage, qui est souvent utilisé à partir de services en ligne.

# VBA

- Le Basic est l'un des premiers et plus influents langages de programmation interprétés. Il est apparu en 1964 et était destiné aux non-informaticiens.
- En 1991 Microsoft lance sa propre version du Basic nommée Visual Basic pour permettre de développer simplement des applications graphiques. Le Visual Basic est obsolète depuis 1998.
- En 1993, Microsoft décide d'intégrer une version du Visual Basic nommée VBA à sa suite Office afin de permettre d'écrire des macros, notamment sous Excel et Access. Le VBA est également obsolète depuis 1998.
- En 2002, Microsoft décide d'implémenter une nouvelle version du Visual Basic avec son .Net Framework, nommée Visual Basic .Net, toujours pour permettre de développer sous Windows. Pour plus de clarté sans doute (sarcasme), Microsoft a décidé de renommer ce nouveau langage en simplement "Visual Basic". Ce nouveau langage n'est pas rétrocompatible avec le Visual Basic ni le VBA.
- La syntaxe du VBA est absolument immonde et aucune personne saine d'esprit ne devrait utiliser VBA, mais les entreprises ne sont pas constituées de gens sains d'esprits. Sachez que si vous travaillez dans une entreprise qui utilise beaucoup d'Excel (toutes) le fait d'utiliser le VBA plutôt que les formules de calcul par cases d'Excel peut vous faire gagner beaucoup d'heures de travail. Notez que vous n'aurez parfois pas d'autres choix que d'utiliser le VBA pour un projet simplement parce qu'en entreprise vous n'aurez pas forcément les droits d'administrateur sur votre PC.
- Pour utiliser le VBA sous Excel il faut activer le mode développeur puis écrire une macro. Deux choses à surveiller : ne confondez pas VBA et Visual Basic (.Net) en recherchant de l'aide sur internet et sachez qu'il est également possible de charger des librairies (DLL) pour y avoir accès dans vos scripts, ce qui peut vous simplifier la vie.



Faire mieux que Java





# C#

- Le C# est un langage de Microsoft apparu en 2000. Il s'agit du langage phare de développement par Microsoft qui était à l'origine pour bonne part un clone de Java.
- La philosophie du C# est en revanche plus proche du C++ en ce sens que le langage est une sorte de fourre-tout avec lequel Microsoft tente de satisfaire tous les publics. C'est une sorte de Java avec plein de fonctionnalités que vous auriez voulu mais du coup assez incohérent.
- Le langage est aujourd'hui cross-platform grâce au .Net Core et un très bon choix pour développer des applications de diverse taille ... si le fait q'il soit maintenu par un GAFAM ne vous dérange pas.

# Scala

- Le langage scala est apparu en 2004 à l'université polytechnique de Lausanne. Le langage fonctionne sur la JVM est parfaitement interopérable avec Java.
- Le langage a connu un certain succès grâce à sa syntaxe épurée et très libre, toutefois il s'est avéré passé un temps que la syntaxe qui avait été proposée par le langage était trop ambiguë pour être parsée correctement et certaines libertés ont du être supprimées depuis.
- Scala a inspiré beaucoup de langages qui ont suivis, dont notamment Kotlin qui est le langage de prédilection pour le développement sous Android.