

Langages et Compilateurs : Introduction

Matthieu Amiguet

2009 – 2010



1 Plan du cours

2 Pourquoi Python ?

3 Compilateurs

Plan

3

- Le langage Python
- Constructions avancées en Python
- Programmation fonctionnelle en Python
 - ↪ Test
- Compilateurs
 - ↪ Projet "Compilateur"
- Introduction à la théorie des langages et des automates
 - ↪ (Test ?)
- Examen

1 Plan du cours

2 Pourquoi Python ?

3 Compilateurs

Qu'est-ce que python ?

5

- Langage de programmation avec
 - gestion de la mémoire
 - gestion des exceptions
- Multi-paradigme
 - Impératif structuré
 - Orienté objet
 - Fonctionnel (dans une certaine mesure)
- Langage qui semble interprété
 - Au fait, compilé à la volée en *bytecode* pour une machine virtuelle...
- Disponible sous Windows, Linux, MacOS, ...
 - Mais aussi implémenté en Java, .Net, et... Python !
- Licence libre et assez permissive

Pourquoi python pour ce cours ?

6

- Langage de haut niveau avec fonctionnalités avancées
 - Structures de données avancées
 - Itérateurs, générateurs, décorateurs
 - "Context managers"
 - Programmation fonctionnelle
 - ...
- ↪ Comparable à C# (ou Java) mais avec des choix de conception assez différents !
- Richesse des bibliothèques
- Excellent langage de prototypage rapide !

Python est "économique"

7

- "Hello, World" en python :

```
print "Hello , world !"
```

- Échanger la valeur de deux variables

```
a,b = b,a
```

- Retourner plusieurs valeurs

```
def pred_succ(x):
    return x-1, x+1

p, s = pred_succ(1)
print "pred_succ(%d) : %d, %d" % (1,p,s)
```

La philosophie "Batteries included"

8

- Un serveur web en trois lignes :

```
import BaseHTTPServer, SimpleHTTPServer
server = BaseHTTPServer.HTTPServer(\
    ('',8080),\
    SimpleHTTPServer.SimpleHTTPRequestHandler\
)
server.serve_forever()
```

La philosophie "Batteries included" (Suite) 9

- Récupération et traitement d'un *feed RSS*

```
import urllib, sys, xml.dom.minidom

address = 'http://www.sebsauvage.net/rss/updates.xml'
document = \
    xml.dom.minidom.parse(urllib.urlopen(address))

for item in document.getElementsByTagName('item'):
    title = item.getElementsByTagName('title')[0].\
        firstChild.data
    print "Title:", title.encode('latin-1','replace')
```

Les limites de python 11

Python n'est pas forcément adapté lorsque...

- La vitesse pure est un critère prioritaire
- On veut un compilateur contraignant
- On ne veut pas diffuser le code source (quoique...)
- On ne veut/peut pas installer l'interpréteur
- ...

Python – pour quels usages ? 10

- Administration système
- Prototypage rapide
- Développement "agile"/orienté par les tests/...
- Développement web (Zope, Django, Web2py ...)
- Outils de recherche scientifique
- Langage d'extension
- Applications pédagogiques
- ...

Qui utilise python ? 12

- Google
 - Participe considérablement au développement du langage
- YouTube
 - "Python is fast enough for our site and allows us to produce maintainable features in record times, with a minimum of developers"
- Industrial Light & Magic
 - "Without it a project the size of Star Wars : Episode II would have been very difficult to pull off."
- Yahoo, NASA, local.ch, ...

↪ <http://www.python.org/about/quotes/>

Un mot sur la version...

13

- Python a récemment passé des versions 2.x aux 3.x
 - beaucoup de changements et *rupture de la compatibilité arrière*
 - En même temps est sortie une version "de transition" 2.6
- Mais
 - certaines bibliothèques utilisées dans ce cours n'ont pas encore été portées vers Python 3.x
 - de nombreux serveurs en sont encore à une version 2.x
 - les documents tiers, tutoriaux, etc. sur le web sont encore beaucoup plus nombreux sur python 2.x
 - ...
- Le cours se basera donc sur python 2.5

- 1 Plan du cours
- 2 Pourquoi Python ?
- 3 **Compilateurs**

Compilateurs

15

- Tout programmeur utilise couramment un (des ?) compilateur(s)
 - Mais on ne sait pas toujours "comment c'est fait dedans"
- Une étude des techniques de compilation permet
 - D'écrire son propre compilateur/interpréteur/...
 - De mieux comprendre cet outil primordial...
 - ... et donc de mieux l'utiliser !
- L'étude des compilateurs nous donnera aussi l'occasion d'aborder
 - Les questions d'analyse de langages
 - La théorie des automates (dont la machine de Turing)
