

# Langages Objets

## Cours 1 : Introduction

Nicolas.Thiery@u-psud.fr

25 novembre 2014

<http://Nicolas.Thiery.name/Enseignement/CCI-L0/>

# Logistique

## Séances

6 x 2 séances de 4h + 1 séance 2h = 50h

Une séance = 1h15 cours + 45mn TD + 2h TP machine

## Site du cours

<http://Nicolas.Thiery.name/Enseignement/CCI-L0/>

Site du cours d'il y a deux ans par Emmanuel Waller :

<http://www.lri.fr/~waller/cours/lo/a2z3/>

# Contrôle des connaissances

## Examen : 3h

- ▶ session 1 : tous documents autorisés
- ▶ session 2 : aucun document autorisé

## Contrôle continu (CC)

- ▶ Assiduité, ponctualité (arrivée, départ)
- ▶ Interrogation 1h vers le bloc 10
- ▶ Notes de TP
- ▶ Éventuellement autres (micro-interrogations, etc.)

## Module

2/3 examen + 1/3 CC

# Objectifs du CCI

## Programmation

- ▶ Être autonome dans : spécifier, concevoir, programmer, tester et maintenir des (petites) applications de qualité
- ▶ Principes des langages de programmation (C), algorithmique, langages objets, projet

## Technologies

- ▶ Utiliser les ordinateurs, avec leur outils matériels et logiciels
- ▶ Architecture, bases de données, réseau, etc.

# Objectif du cours

Développer une petite application métier : SimSavane

Demande du client : « Je voudrais une application permettant de visualiser l'évolution d'un écosystème en zone de Savane »

# Compétences

## Modélisation objet

- ▶ Objet, classe, méthode
- ▶ Encapsulation
- ▶ Héritage
- ▶ Polymorphisme

## Principes et patrons de conception

- ▶ DRY
- ▶ Modèle-Vue-Contrôleur

## Génie logiciel

- ▶ Développement incrémental
- ▶ Tests !!! Automatisation !!!

# Technologies

## Langage Java

- ▶ Variables et types de base, tableaux
- ▶ Instructions et expressions
- ▶ Contrôle de flux : conditionnelles (if), boucles (for, while)
- ▶ Fonctions, fonctions récursives

## Applications interactives

- ▶ Graphiques
- ▶ Multitâche

## Environnement de programmation

- ▶ Éditeur de texte + Shell + Makefile
- ▶ Eclipse

# Pourquoi Java

## Quelques avantages

- ▶ Langage de programmation courant
- ▶ Cadre strict pour débiter :
  - ▶ Langage compilé typé statiquement
  - ▶ Messages d'erreurs à l'exécution lisibles
- ▶ Orienté objet



# Pourquoi Java

## Quelques avantages

- ▶ Langage de programmation courant
- ▶ Cadre strict pour débiter :
  - ▶ Langage compilé typé statiquement
  - ▶ Messages d'erreurs à l'exécution lisibles
- ▶ Orienté objet
- ▶ Bibliothèque standard complète :  
structures de données, applications graphiques et interactives, réseaux, ...
- ▶ Environnement de programmation confortable

# Pourquoi Java

## Quelques avantages

- ▶ Langage de programmation courant
- ▶ Cadre strict pour débiter :
  - ▶ Langage compilé typé statiquement
  - ▶ Messages d'erreurs à l'exécution lisibles
- ▶ Orienté objet
- ▶ Bibliothèque standard complète :  
structures de données, applications graphiques et interactives, réseaux, ...
- ▶ Environnement de programmation confortable

## Cela dit ...

Ailleurs, il m'arrive d'enseigner C++ ou Python aux débutants !

# Premiers pas en Java

Voir les premiers exercices du TD.

# Bibliographie

- ▶ Best Practices in Scientific Computing
- ▶ Claude Delannoy, Programmer en Java, deuxième édition mise à jour, Eyrolles 2002
- ▶ Claude Delannoy, Exercices en Java, Eyrolles, 2001
- ▶ Java in a nutshell, O'Reilly, au moins 3ème édition, 2000
- ▶ <http://docs.oracle.com/javase/tutorial/>
- ▶ [http://fr.wikipedia.org/wiki/Extreme\\_programming](http://fr.wikipedia.org/wiki/Extreme_programming)
- ▶ ...