

# Contexte national de création du DIU

## Demande du ministère

- Cadre de la réforme Blanquer du baccalauréat
- Création de la discipline informatique au lycée et montée en puissance de l'enseignement de l'informatique en primaire et collège
- Nécessite la formation de professeurs pour la spécialité "Numérique et Sciences Informatiques" (NSI) de Première et Terminale
  - Programme plus ambitieux que ISN
  - 4h hebdomadaire en classe de Première
  - 6h hebdomadaire en classe de Terminale

## Objectifs de la formation

- Accompagner les futurs enseignants de NSI
- Reconnaître leur investissement par un DIU

# Cadrage national

- Formation de 175h sur 2 ans dont 125h de présentiel et 50h à distance
- Décomposée en 5 blocs thématiques de 35h (25h prés. + 10h dist.)
- 3 blocs la première année puis 2 blocs la deuxième année
  - Première année : préparation au programme de Première
  - Deuxième année : préparation au programme de Terminale
- Possibilité de changer d'académie entre les deux années

# Paris 13 : centre secondaire

## National

novembre **Programme de NSI**

janvier **Création du DIU**

Avril **Premiers stages des universités référentes**

## Paris 13

Mars **Participation mentionnée**

Avril **Choix des responsables de blocs**

Juin **Premiers stages**

(Pression du rectorat pour participer)

# Mise en place : Des couacs dans tous les sens

## National

- **Opposition politique**
- **Départements surchargés**
- **Manques de ressources**
- **Annulations**

## Paris 13

- **Démission d'un responsable**
- **Lenteur administrative**  
(formation pas encore validés par les instances de l'université...)
- **Réorganisation complète il y a moins de deux semaines...**

# Organisation de la formation

Équipe pédagogique : Plus d'une vingtaine d'intervenants, de l'IUT et de l'institut Galilée Responsabilité des 5 UE (blocs)

- Organisation : Camille Coti (MCF IUT)
- Bloc 1 Représentation des données et programmation : Flavien Breuvert (MCF Galilée)
- Bloc 2 Algorithmique : Kais Klai (MCF Galilée)
- Bloc 3 Architecture matérielle et robotique, systèmes et réseaux : Christophe Fouqueré (Pr Galilée)
- Bloc 4 Programmation avancée et bases de données : Flavien Breuvert
- Bloc 5 Algorithmique avancée : Christophe Fouqueré

# Évaluation

Préparation de séquences :

- par binômes,
- une séquence par binôme pour les blocs 1 et 2,
- une séquence par binôme pour chacun des autres blocs,
- but : 24 séquences de prêts par an.

Notation : uniquement une validation.

# Calendrier de la formation

- 11,12,13,18,19,20 juin : Blocs 1 et 2 en panaché
- 2, 3, 4 juillet : Bloc 3
- Toussaint : Bloc 4
- Décembre : Bloc 5

# Site web des blocs 1 et 2

<https://lipn.univ-paris13.fr/~breuvar/diu-eil/bloc1>

Responsable : Flavien Breuvar (breuvar@lipn.fr)

Bientôt sur cette page :

- contenu des blocs
- découpage en séquences
- supports pédagogiques

# Déroulé d'une journée type

9h-12h : Cours/TD ou Cours/TP

Informations essentielles sur un des sujets clé de la formation

13h-17h : TD/TP

Un ou deux TD ou TP illustrant ou complétant le cours du matin

17h-18h : Discussions didactique

Discussions avec divers intervenants de l'université autour d'aspects pédagogiques

# Pourquoi Python ?

- Un langage est souvent créé dans une perspective d'application,

# Pourquoi Python ?

- Un langage est souvent créé dans une perspective d'application,
- un **langage professionnel** pour une application donnée est souvent **le pire langage pour apprendre** à faire cette application, en effet il permet de prendre en compte de **nombreuses subtilités que l'on souhaiterait cacher à un débutant**.

# Pourquoi Python ?

- Un langage est souvent créé dans une perspective d'application,
- un **langage professionnel** pour une application donnée est souvent **le pire langage pour apprendre** à faire cette application, en effet il permet de prendre en compte de **nombreuses subtilités que l'on souhaiterait cacher à un débutant**.
- les **langages pédagogiques** existent mais disposent de **peu de bibliothèques** et ont très mauvaise presse auprès des étudiants cherchant à ajouter des lignes sur leur CV...

# Pourquoi Python ?

- Un langage est souvent créé dans une perspective d'application,
- un **langage professionnel** pour une application donnée est souvent **le pire langage pour apprendre** à faire cette application, en effet il permet de prendre en compte de **nombreuses subtilités que l'on souhaiterait cacher à un débutant**.
- les **langages pédagogiques** existent mais disposent de **peu de bibliothèques** et ont très mauvaise presse auprès des étudiants cherchant à ajouter des lignes sur leur CV...
- **Python n'est PAS FAIT pour faire de l'algorithmique**, ni de la gestion de projet, c'est pour ça que c'est un candidat potentiel pour l'enseigner...

# Caractéristiques (rapides) de Python

- Un langage de script : fait pour manipuler des codes écrits en C
- Bibliothèques extensives
- Langage interprété ( $\neq$  langage compilé)
- Typage Dynamique ( $\neq$  typage statique)
- Gestion automatique de la mémoire (garbage collector)
- Orientation objet + quelques aspects fonctionnels
- Indentation significative (moins de parenthèses)
- Structures de données par défaut (lent mais facile)
- Méta-programmation riche (manipulation de l'interpréteur)