

# Projet de programmation (IK3) : TP n°1

Semaine du 20 septembre 2010

Ce TP doit être entièrement réalisé avec Eclipse. Vous utiliserez un seul *workspace* pour toute la séance et chaque question fera l'objet d'un projet dans ce *workspace*.

## 1 Entrées/sorties, types de bases et structures de contrôle

Pour ce cours, les entrées/sorties dans la console se feront à l'aide de la classe `DeugConsole` que nous vous fournissons. La création d'un objet de ce type se fait par exemple ainsi :

```
DeugConsole console = DeugConsole.console();
```

Il faut pour cela avoir importé le type `deug.DeugConsole`. Une fois cet objet obtenu, il est possible d'utiliser ses fonctions membres pour effectuer des entrées/sorties sur les types de bases (`int`, `char` et `String`). Pour chaque exercice de cette série, vous n'utiliserez qu'une seule classe, dont toutes les fonctions seront **static**.

**Tests et types de bases** : écrire un programme qui lit sur l'entrée standard trois entiers, puis les affiche du plus petit au plus grand ; faire de même avec des chaînes de caractères.

**Tableaux et boucles** : écrire un programme qui lit sur l'entrée standard les noms et notes d'un certain nombre d'étudiants et qui les réécrit classés par ordre alphabétique en utilisant un *tri à bulles*.

**Fonctions récursives** : écrire un programme qui calcule la suite de Fibonacci à l'aide d'une boucle et un autre qui la calcule à l'aide d'une fonction récursive ; comparer et commenter les temps d'exécution de ces deux programmes.

**Tris** : modifier le programme de tri pour qu'il utilise un algorithme de tri plus efficace (au choix : tri fusion, tri rapide ou tri par insertion).

## 2 Programmation orientée objet et communications réseaux

### Dérivation de fonctions

En se basant sur une *interface* pour une classe calculant une fonction, écrire une classe qui calcule la dérivée en un point d'une fonction qui a été passée à son constructeur.

### Serveur echo

Le protocole echo décrit un service de communication dans lequel un serveur renvoie telles quelles aux clients qui lui sont connectés les données que ceux-ci envoient. En utilisant la classe `DeugSocket` fournie, écrire un programme qui, lorsqu'il est lancé sans argument, joue le rôle du serveur et, lorsqu'il reçoit un nom d'hôte en argument, se connecte au serveur spécifié, lui transmet les données entrées sur la console et affiche celles qu'il reçoit du serveur.

### Client HTTP

À l'aide de la description du protocole HTTP<sup>1</sup>, écrire un programme qui demande l'adresse d'une page web à l'utilisateur, se connecte au serveur gérant cette page, effectue la requête HTTP nécessaire pour obtenir la page et affiche son code HTML à l'utilisateur.

---

1. Par exemple là : [http://en.wikipedia.org/wiki/Hypertext\\_Transfer\\_Protocol#Example\\_session](http://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol#Example_session).