

## **Interprétation du langage.**

Avec l'aide d'un **interprète**, nous communiquons avec une chinoise sans connaître sa langue :

Bonjour Mademoiselle Li !

早, 李太太, 您早!



De la même manière, nous pourrions communiquer avec l'ordinateur sans connaître son langage. Nous allons employer l'ordinateur lui-même (ou plus exactement un programme appelé «**interprète**» ou «**compilateur**» selon le cas) pour effectuer la traduction du langage que vous utilisez dans celui de l'ordinateur :

Bonjour eMAC !

0101011000100010001  
0100110011101110110  
010011101111011101...

...



NB : «communiquer avec l'ordinateur» est une notion toute relative : l'eMAC ne comprend rien à ce que je lui raconte. Il se borne à exécuter. Lui dire bonjour est donc inutile... Le test habituel est «Hello world !», formule consacrée qui démontre que nous allons communiquer à *partir de notre application* avec le monde, mais aussi que nous traitons notre programme comme le début d'une *entité autonome qui parlerait en son nom propre*.

Bonjour eMAC !



```
public class Hello {  
    public static void main(String[] args)  
    {  
        System.out.println(«Bonjour eMac !»);  
    }  
}
```



```
0101011000100010001  
0100110011101110110  
010011101111011101...
```



...



Bonjour eMAC !



**Français :**



**Processing :**

```
size(200, 200);  
background(255);  
PFont fontA = loadFont(«HelveticaNeue-  
48.vlw»);  
textFont(fontA, 14);  
int x = 50;  
fill(0);  
text(«Bonjour eMac !», x, 100);
```



**Java :**

```
public class Hello {  
    public static void main(String[] args)  
    {  
        System.out.println(«Bonjour eMac !»);  
    }  
}
```



**Assembleur :**

```
0101011000100010001  
0100110011101110110  
010011101111011101...
```



**Langage-machine :**

```
0101011000100010001  
0100110011101110110  
010011101111011101...
```



...



**PROCESSING** est un environnement qui simplifie la création de programmes dans un langage plus extensible, plus puissant, mais plus complexe : Java.

Si vous voulez savoir à quoi ressemblerait votre programme si vous l'aviez programmé directement dans Java, ouvrez le fichier .java qui se trouvera à l'intérieur du dossier "applet" généré lorsque vous exporterez votre création dans PROCESSING afin de la publier en ligne (fonction "export", qui produit un ensemble de fichiers dont une page HTML). C'est une bonne manière de comprendre ce qui se passe.

Processing fonctionne grâce à Java et ses outils.

Concernant le schéma de la page précédente, j'ai volontairement simplifié les étapes qui mènent de **PROCESSING** au langage-machine. Java est un langage **compilé** : il est traduit en langage machine à l'aide d'un **compilateur**. Toutefois, contrairement aux langages compilés traditionnels, pour lesquels le compilateur crée un fichier binaire directement exécutable par un processeur donné (c'est-à-dire un fichier binaire contenant des instructions spécifiques à un processeur), le code source Java est compilé en un langage intermédiaire (appelé **pseudo-code** ou **bytecode**) dans un fichier portant le même nom que le fichier source à l'exception de son extension (.class).

Pour en savoir plus :

<http://www.commentcamarche.net/java/javacarac.php3>