

Langage compilé langage interprété

JM Routoure : LI sciences pour l'ingénieur
Université de Caen Basse-Normandie

Pourquoi programmer

- Un ordinateur seul avec son système d'exploitation ne sait rien faire => nécessité de faire tourner des programmes pour
 - faire des taches bien particulières : traitement de texte, dessin, CAO...
 - automatiser des taches
 - ...
- Nécessité de programme qui peuvent être de 2 types : compilés ou interprétés

programme compilé

Code source

Fichier texte

ne dépend pas du
système
d'exploitation



Compilation
= traduction

Objet =
exécutable

Fichier binaire contenant
Code binaire
compréhensible par le
microprocesseur

dépend du système
d'exploitation et du
matériel

Exemple de langage de programmation basé sur
un compilateur : C, C++, java,
Intérêt : rapidité et optimisation du code
Désavantage : complexité

programme interprété

Code source

Interprétation = chaque ligne correspond à une commande qui est exécuté en “lisant” le fichier texte

Fichier texte

ne dépend pas du
système
d'exploitation

Exemple de langage de programmation basé sur un interpréteur : shell, BASIC, langage de macro
Intérêt : simplicité
Désavantage : lenteur

Travail à effectuer

- “hello world” en C et en shell

Réaliser un programme qui affiche hello world et s'arrête !

Structure d'un programme source C

```
#include <bibliotheque> ← insertion des  
                           bibliothèques  
  
int main(){  
    ;  
    return (0); ← programme principal  
}
```

compilation : `gcc -o mon_executable fichier_source.c`

```
#include <stdio.h>
#include <stdlib.h>

int main(){
    printf("Hello world");
return (0);
}
```

**Compilation d'un
logiciel libre à montrer !**