



Bibliothèques logicielles (modules)

◊ Une *bibliothèque logicielle* permet de regrouper différentes fonctions afin de les réutiliser sans avoir besoin de réécrire le code. Les fonctions d'une même bibliothèque concernent en général un même domaine (mathématiques, nombres complexes, matrices, représentations graphiques, etc.) Ces fonctions doivent être soigneusement documentées et testées. Idéalement, on doit pouvoir utiliser les fonctions d'une bibliothèque sans savoir comment elles ont été écrites mais uniquement en connaissant leur documentation. En PYTHON, les bibliothèques sont appelées *modules*. Rappelons que pour utiliser un module à l'intérieur d'un programme, on utilise la commande **import** sous l'une des trois formes suivantes :

```
import math
```

On devra utiliser les fonctions sous la forme **math.sin**, **math.cos**, etc.

```
from math import sin, cos
```

On importe seulement les fonctions **sin** et **cos** du module **math**, on pourra utiliser ces fonctions directement (sans écrire **math.**)

```
from math import *
```

On importe *toutes* les fonctions du module **math**

⚠ Il est déconseillé d'utiliser systématiquement la dernière forme car des modules différents peuvent avoir des fonctions avec le même nom.

```
"""
Mini module de maths
"""
pi=3.14
def abs(x):
    """
    Calcule la valeur
    absolue de x:float
    """
    if x>=0:
        return x
    else:
        return -x
```

◊ Un module PYTHON est simplement un fichier **.py**. On donne ci-contre l'exemple d'un fichier **minimath.py** constituant un module mathématique comportant uniquement une valeur (très) approchée de π et une fonction de calcul de la valeur absolue. La première chaîne de caractères est la documentation du module.

◊ Pour utiliser ce module dans un programme :

```
import minimath
print minimath.abs(-2)
```