

# Fonctions imbriquées et à plusieurs arguments

## NUMWORKS


Cette séance a pour objectifs de programmer en utilisant une fonction à plusieurs arguments, de créer une fonction qui appelle une autre fonction et de découvrir l'utilité d'une fonction.

### 1 Découverte d'une fonction à deux arguments

(a) Nous allons commencer par créer un nouveau script nommé `activite2.py`. Voici une fonction nommée `maximum` qui prend en entrée deux nombres et qui renvoie le maximum des deux valeurs.

```
deg PYTHON
from math import *

def maximum(a, b):
    if a > b:
        return a
    else:
        return b
```

La saisir dans le script `activite2.py` puis faire **Exécuter le script** dans le menu à droite du titre du script. La fonction ainsi créée est disponible dans le menu de la touche . Appeler cette fonction avec différentes valeurs, par exemple `maximum(12,15)`, `maximum(6,-8)`, ...

(b) Ecrire dans le même script une fonction que vous nommerez `minimum` qui prend en entrée deux nombres et qui renvoie le minimum des deux valeurs.

### 2 Découverte d'une fonction à plus que deux arguments

(a) Ecrire dans le même script une fonction que vous nommerez `maximum3` qui prend en entrée trois nombres et qui renvoie le maximum des trois valeurs.

(b) Ecrire dans le même script une fonction que vous nommerez `maximum4` qui prend en entrée quatre nombres et qui renvoie le maximum des quatre valeurs.

**Aide** : Vous pouvez utiliser votre fonction `maximum` et vous appuyer sur un schéma

(c) Ecrire dans le même script une fonction que vous nommerez `maximum8` qui prend en entrée huit nombres et qui renvoie le maximum des huit valeurs en utilisant la fonction `maximum4`.

**Aide** : faire un schéma avec le résultat de chaque appel aux fonctions