

Arrondir un nombre décimal, c'est le transformer en un entier. Il existe plusieurs manières de calculer des arrondis et nous en verrons deux dans ce cours :

- l'arrondi à l'entier **inférieur**,
- l'arrondi à l'entier **supérieur**.

Par exemple, si on a une température de 12.3 degrés alors les deux arrondis ci-dessus vont donner les valeurs 12 (entier inférieur) et 13 (entier supérieur).

En C++ pour calculer des arrondis il faut tout d'abord avoir inclu la bibliothèque *cmath* :

```
#include <cmath>
```

Pour calculer les arrondis on fait ainsi :

```
// Entier inférieur (partie entière)
double arrondiInf = floor(12.3);
cout << arrondiInf << endl;

// Entier supérieur
double arrondiSup = ceil(12.3);
cout << arrondiSup << endl;
```

```
↳ 12
   13
```

Attention si le nombre est négatif !

```
// Entier inférieur (partie entière)
double arrondiInf = floor(-12.3);
cout << arrondiInf << endl;

// Entier supérieur
double arrondiSup = ceil(-12.3);
cout << arrondiSup << endl;
```

```
↳ -13
   -12
```

Selon la situation, ce sera à vous de déterminer de quelle fonction d'arrondi vous avez besoin.

Comment se souvenir des noms de ces fonctions ?

- `floor` comme "floor" la traduction en anglais du mot "sol" : le sol est en bas, donc on arrondi à l'entier inférieur
- `ceil` comme "ceiling" la traduction en anglais du mot "plafond" : le plafond est en haut, donc on arrondi à l'entier supérieur