

Pour aider à organiser les noms lorsque l'on inclut beaucoup de bibliothèques, le C++ propose d'utiliser des *espaces de noms*. En écrivant :

```
namespace espaceDeNoms
{
    // Noms
}
```

une ou plusieurs fois dans le code source, les noms définis dans l'espace devront être préfixés par `espaceDeNoms::` pour être utilisés. On mettra des noms dans l'espace en plusieurs fois :

```
namespace calculsPratiques
{
    template <typename T>
    T min(const T& a, const T& b)
    {
        if (a < b)
            return a;
        return b;
    }
}

namespace calculsPratiques
{
    template <typename T>
    T max(const T& a, const T& b)
    {
        if (a > b)
            return a;
        return b;
    }
}

int lePlusGrand = calculsPratiques::max(nb1, nb2);
```

Le `::` est appelé « opérateur de résolution de portée ». Pour éviter de recopier le préfixe, on peut utiliser l'instruction `using` :

```
using calculsPratiques::min;
using calculsPratiques::max;
```

Les noms de la bibliothèque standard du C++ sont tous placés dans l'espace de noms *std*. Ainsi, dans nos programmes, nous écrivons :

```
using namespace std;
```

En écrivant `using namespace`, on récupère directement tous les noms d'un espace de noms.

Il ne vous sera jamais avantageux de créer des espaces de noms dans nos exercices, puisque nous écrivons des programmes courts avec relativement peu de noms ; les espaces ne feraient qu'alourdir. Cette notion peut toutefois être intéressante à connaître et à comprendre pour maîtriser le langage C++ ; elle se rapproche de la notion de *paquetage* de certains langages.