Manuel

La classe *IntervalSet* a pour rôle de gérer le stockage, la manipulation et l'affichage d'un nombre arbitraire d'intervalles disjoints ajoutés par l'utilisateur.

Manuel d'utilisation des méthodes objet

`Display ( ) const`

Description :

Réalise l'affichage sur la sortie standard du contenu de l'*IntervalSet*.
L'affichage se fait sous la forme suivante :
{ [a, b] … [y, z] }
Où [a, b] est le premier intervalle, [y, z] est le dernier et les points de suspension représentent l'ensemble des intervalles contenus dans l'objet et compris entre ces deux-là.

Valeur de retour :

Aucune (*void*).

Exemple d'utilisation :

```
#include "IntervalSet.h"

int main ( )

{
	IntervalSet is;
	is.Add(0,0);
	is.Add(2,3);
	is.Add(5,7);
	is.Add(-57,-24);
	is.Add(-2,-1);
	is.Display();
}
```

Ce qui produit l'affichage :

```
{[-57,-24], [-2,-1], [0,0], [2,3], [5,7]}
```

`IntervalSet& Intersection ( const IntervalSet& is ) const`

Description :

Renvoie une référence vers un nouvel objet *IntervalSet* constitué de l'intersection de l'*IntervalSet* « *this* » sur lequel est appelée la méthode et de l'*IntervalSet* « *is* » passé en argument.
Les objets originaux *this* et *is* ne sont pas affectés par l'appel à cette méthode. Le nouvel *IntervalSet* contient ses propres intervalles et n'est pas affecté par les modifications ultérieures des objets originaux.

Cette méthode alloue de la mémoire pour l'objet dont la référence est retournée, **mémoire qui devra être libérée par l'utilisateur** lorsque cet objet ne sera plus utilisé.

Valeur de retour :

Un *IntervalSet* alloué sur le tas contenant l'intersection des deux *IntervalSet* originaux.

Exemple d'utilisation :

```
#include "IntervalSet.h"

int main ( )

{
	IntervalSet is;
	is.Add(0,1);
	is.Add(2,3);
	is.Add(4,8);
	is.Add(10,15);
	is.Add(9,9);
	is.Add(45, 47);
	is.Display();

	IntervalSet is2;
	is2.Add(1,2);
	is2.Add(3,5);
	is2.Add(10,18);
	is2.Add(42, 47);
	is2.Add(22,23);
	is2.Display();

	cout << "Intersection : " << endl;

	IntervalSet& is3 = is.Intersection (is2);
	is3.Display();

	delete &is3;
}
```

Ce qui produit l'affichage :

```
{[0,1], [2,3], [4,8], [9,9], [10,15], [45,47]}
{[1,2], [3,5], [10,18], [22,23], [42,47]}
Intersection :
{[1,1], [2,2], [3,3], [4,5], [10,15], [45,47]}
```