

TP de programmation fonctionnelle et logique

TP 1 : listes et récursivité

Fonctions classiques sur les ensembles triés

On considère des ensembles triés représentés par des listes. Exemple

```
#let l1 = [1;3;5;7;9];;  
val l1 : int list = [1; 3; 5; 7; 9]
```

```
#let l2 = [2;3;6;8;9];;  
val l2 : int list = [2; 3; 6; 8; 9]
```

Écrire des fonctions réalisant :

1. L'intersection; Exemple :

```
#intersection l1 l2;;  
- : int list = [3; 9]
```

2. L'union; Exemple :

```
#union l1 l2;;  
- : int list = [1; 2; 3; 5; 6; 7; 8; 9]
```

3. La différence; Exemple :

```
#différence l1 l2;;  
- : int list = [1; 5; 7]
```

4. La différence symétrique; Exemple :

```
#différence_symétrique l1 l2;;  
- : int list = [1; 2; 5; 6; 7; 8]
```

5. Le produit cartésien (résultat donné dans l'ordre lexicographique). Exemple :

```
#produit_cartésien l1 l2;;  
- : (int * int) list =  
[(1, 2); (1, 3); (1, 6); (1, 8); (1, 9); (3, 2); (3, 3); (3, 6); (3, 8);  
(3, 9); (5, 2); (5, 3); (5, 6); (5, 8); (5, 9); (7, 2); (7, 3); (7, 6);  
(7, 8); (7, 9); (9, 2); (9, 3); (9, 6); (9, 8); (9, 9)]
```