

# Lambda calcul et programmation fonctionnelle

## TD 1

### Exercice 1

Voici ci-dessous des fonctions écrites en Haskell. Selon vous, quel sont leurs types (ou leurs signatures, si vous préférez) ? Il peut parfois y avoir plusieurs réponses possibles.

```
inverse x = 1/x
```

```
charToString c = [c]
```

```
second l = l !! 1
```

```
moyenne a b c = (a + b + c) / 3
```

```
moyenne2 notes = (sum notes) / (length notes)
```

### Exercice 2

Selon vous, quel est le type de la fonction `sum` utilisée ci-dessus ?

### Exercice 3

Comme dans tous les langages, Haskell permet de renvoyer des résultats différents en fonction d'une condition, en utilisant le mot-clé `if`. Voici un exemple d'utilisation du `if` :

```
jugement notes =  
    if (moyenne notes) >= 10  
        then "Bien ouéj"  
        else "T'es trop nul"
```

En Haskell :

- `if` renvoie un résultat (car Haskell est fonctionnel : tout ce qu'on écrit renvoie toujours une valeur)
- il y a toujours un `else` (car quand la condition est fausse, quel résultat doit-on renvoyer ?)
- le type du résultat renvoyé par la branche `then` est toujours le même type que celui renvoyé par la branche `else` (car sinon, quel serait le type de l'expression ?)

Selon vous, quel est le type de la fonction `jugement` ?

### Exercice 4

Écrivez une fonction `abs` qui calcule la valeur absolue d'un entier passé en paramètre. N'oubliez pas la signature avant d'écrire la fonction.

Écrivez une fonction `max` qui renvoie le plus grand nombre entre deux entiers passés en paramètre.

Écrivez une fonction `second2` qui renvoie le second élément d'une liste d'entiers passée en paramètre, mais uniquement s'il y a au moins deux éléments dans la liste. Dans le cas contraire, on renverra -1 (par convention).

Écrivez une fonction `moyenne3` qui renvoie la moyenne d'une liste de notes passée en paramètres. Mais attention : parfois la liste de notes peut être vide. Dans ce cas, on renverra -1 (par convention).

Écrivez une fonction `max3` qui renvoie le plus grand de trois nombres entiers passés en paramètre.

Écrivez une fonction `jugement2` qui renvoie une chaîne de caractères émettant un jugement de valeurs en fonction des notes obtenues par un étudiant. Si l'étudiant a eu la moyenne mais n'a pas eu au moins 15, on renvoie « Pas mal ». S'il a eu 15 ou plus, on renvoie « Bravo champion ». S'il a eu moins de 10, on renvoie « Dommage ». Et enfin, s'il n'a pas eu de note, on renvoie « Absent ».

Écrivez une fonction `estVide` qui renvoie `True` si la liste passée en paramètre est de longueur 0, `False` dans le cas contraire.

### Exercice 5

Écrivez une fonction `maxList` qui prend en paramètre une liste d'entiers et qui renvoie la plus grande valeur présente dans la liste. On considèrera que la liste n'est pas vide au départ.

Écrivez une fonction `maxList2` qui prend en paramètre une liste d'entiers et qui renvoie la plus grande valeur présente dans la liste. Cette fois, on considère que la liste est peut-être vide. Si c'est le cas, on renvoie -1, par convention.

Question : quel sera la valeur de `maxList2 [-3, -1, -5, -2]` ? Qu'en concluez-vous ?