

LIFAP5 – Programmation fonctionnelle pour le WEB

TD2 – programmation fonctionnelle en javascript

Licence informatique UCBL – Printemps 2017–2018

Exercice 1 : Curryfication et application partielle

1. Donner le type puis écrire une fonction `c_add` qui prend un nombre en argument et renvoie une fonction qui prend un autre nombre en argument et renvoie la somme des deux.
2. Donner le type puis écrire une fonction `curry2` qui prend en argument une fonction `f` à deux arguments et renvoie sa version curryfiée, c'est-à-dire une fonction qui prend le premier argument et renvoie une fonction qui prend le deuxième argument qui renvoie le résultat de `f` appliquée aux deux arguments.
3. On définit `let plus = (x,y)=> x + y;`. Utiliser `curry2` pour proposer une nouvelle définition de `c_add`.
4. Soit la définition de `let combine = f => f(2)* f(3);`. Calculer `combine(c_add(5))`.
5. Donner le type puis écrire une fonction `uncurry2` qui soit l'inverse de `curry2`, c'est-à-dire qui a les propriétés `uncurry2(curry2(f)) = f` et `curry2(uncurry2(f)) = f`.
6. Traduire les définitions de `uncurry2` et de `curry2` en λ -calcul
7. On rappelle la définition de $\mathbf{K} = \lambda x. \lambda y. x$, prouver que le curryfié de π_1 est \mathbf{K} . Quelle expression du λ -calcul correspond au curryfié de π_2 .
8. Prouver les propriétés qui unissent `curry2` et `uncurry2`.

Exercice 2 : Curryfication appliquée à `map()`

On rappelle que la méthode `Array.prototype.map(f)` crée un nouveau tableau composé des images des éléments du tableau appelant par la fonction `f` donnée en argument. Pour un tableau `A = [v_0, v_1, ..., v_n]`, `A.map(f)` renvoie `[f(v_0), f(v_1), ..., f(v_n)]` et laisse `A` inchangé.

1. Donner le type puis écrire une fonction `c_map` qui prend en argument une fonction `f` et renvoie une fonction qui prend un tableau `t` et applique `f` à chaque élément de `t`.
2. Utiliser `c_add` et `c_map` pour définir une fonction qui ajoute 3 à tous les éléments d'un tableau sans utiliser `function` et `=>`.
3. Soit un tableau `t`. Expliquer ce que calcule la fonction `c_map(f => f(3))(t)`
4. Soit `A` un tableau qui contienne des objets représentant des personnes avec les champs `nom`, `prénom` et `age`. Définir une fonction qui transforme `A` en un tableau ne comportant que les ages. Le faire également en utilisant la fonction `let prj = c => o => o[c];`.