

Exercices 36

Écrire une fonction `trouve(p, t)` qui reçoit en arguments une fonction `p` (pour propriété) et un tableau `t` et renvoie le premier élément `x` de `t` tel que `p(x)` vaut `True`. Si aucun élément de `t` ne satisfait `p`, alors la fonction renvoie `None`.

Écrire une fonction `filter(p, t)` qui reçoit en arguments une fonction `p` (pour propriété) et un tableau `t` et renvoie tous les éléments `x` de `t` tel que `p(x)` vaut `True`. Si aucun élément de `t` ne satisfait `p`, alors la fonction renvoie un tableau vide.

Exercice 37

Écrire une fonction `applique(f, t)` qui reçoit en arguments une fonction `f` et un tableau `t` et renvoie un nouveau tableau, de même taille, où la fonction `f` a été appliquée à chaque élément de `t`. Le faire avec et sans la notation par compréhension.

Exercice 39 :

Écrire une fonction `double(f)` qui reçoit une fonction `f` en argument et renvoie une fonction qui applique deux fois de suite la fonction `f` à son argument.

Que vaut `double(double(lambda x : x:x*x) (2))` ? (Trouver le résultat de tête, puis lancer le calcul pour le vérifier).

Exercice 40 :

Écrire une fonction `compose(f, g)` qui reçoit en arguments deux fonctions `f` et `g` et renvoie leur composition, c'est à dire la fonction `h` telle que, pour tout `x`, `h(x)` est égale à `f(g(x))`.