

Feuille 3bis : Exercice complémentaire

À ne faire que si vous avez fini les 3 premières feuilles y compris le calcul de $fib(n)$ en temps logarithmique.

On souhaite manipuler des couples d'entiers. Pour cela on utilise le type :

```
type couple = C of int * int
```

1. Écrire le constructeur `make_couple` qui permet de construire un couple à partir de deux entiers.
2. Écrire les accesseurs `couple_fst` (resp. `couple_snd`) permettant d'accéder au premier (resp. deuxième) élément d'un couple de type couple.
3. Écrire la fonction `couple_sum` qui retourne la somme des deux composantes d'un couple.
4. Écrire la fonction `couple_add` qui prend en paramètre deux couples et retourne un couple dont la première (resp. deuxième) composante est la somme des premières (resp. deuxièmes) composantes des deux couples.
5. Écrire une fonction `pair_of_couple` qui retourne la paire (sans constructeur) des deux éléments du couple.

```
# make_couple 1 2;;
- : couple = C (1, 2)
# couple_fst (make_couple 1 2);;
- : int = 1
# couple_snd (make_couple 1 2);;
- : int = 2
# couple_sum (make_couple 1 2);;
- : int = 3
# couple_add (make_couple 1 2) (make_couple 3 4);;
- : couple = C (4, 6)
# pair_of_couple (make_couple 1 3);;
- : int * int = (1, 3)
```

On décide de changer le type couple pour

```
type couple = bool -> int
```

Un couple est représenté par une fonction qui s'applique à un booléen. Appliquée à `true` (resp. `false`), la fonction retourne la première (resp. deuxième) composante du couple.

6. Réécrire le constructeur `make_couple` et les deux accesseurs `couple_fst` et `couple_snd` avec ce nouveau type.
7. Les fonctions `couple_sum`, `couple_add` et `pair_of_couple` que vous avez écrites restent-elles valables avec le nouveau type ?

Exemples :

```
# make_couple 1 2;;
- : bool -> int = <fun>
# couple_fst (make_couple 1 2);;
- : int = 1
# couple_snd (make_couple 1 2);;
- : int = 2
```

```
# couple_sum (make_couple 1 2);;
- : int = 3
# couple_add (make_couple 1 2) (make_couple 3 4);;
- : bool -> int = <fun>
```