

Correction du DS 1

Option informatique, première année

Julien REICHERT

Question 1 : Pour un tableau, `t.(0)` ; pour une chaîne, `s.[0]` ; pour une liste, `List.hd l` ou par un filtrage.

Question 2 : On utilise `&&`.

Question 3 : Pour un tableau, `t.(i) <- x` si `i` est entre 0 (inclus) et la taille de `t` (exclue) et si `x` est du type des éléments de `t` ; pour une chaîne, c'est impossible ; pour une liste c'est impossible, et pas seulement parce que l'accès direct à un élément est interdit ; pour une référence, on écrit `r := x` sous réserve que le type soit compatible.

Question 4 : `print_string "Bonjour"; print_int n`

Question 5 : Le mot-clé `let` introduit une ou plusieurs (re)création(s) de valeur, qui est(sont) fermée(s) par deux points-virgules si la création est sur toute l'expression, sinon par `in`, ce qui est le cas dans les boucles.

Question 6 : Le balisage des boucles est `do ... done`.

```
(* exo1 : unit -> float array *)
```

```
let exo1 () =  
  let tab = Array.make 42 0. in  
  for i = 1 to 42 do  
    tab.(i-1) <- 1. /. float_of_int i  
  done; tab;;
```

```
(* produit_ref : int ref -> int ref -> int ref *)
```

```
let produit_ref a b = ref (!a * !b);;
```

```
(* somme_liste : int list -> int *)
```

```
let rec somme_liste l = match l with  
| [] -> 0  
| a::q -> a + somme_liste q;;
```

```
(* Les trois fonctions de retrait : 'a list -> 'a -> 'a list *)
```

```
let rec retrait_un l elt = match l with  
| [] -> [] (* mais cas impossible a priori *)  
| a::q -> if a = elt then q else a::(retrait_un q elt);;
```

```
let rec retrait_tous l elt = match l with  
| [] -> []  
| a::q -> if a = elt then retrait_tous q elt else a::(retrait_tous q elt);;
```

```
let retrait_tous_erreurs l elt = if List.mem elt l then retrait_tous l elt else failwith "Pas dedans";;
```

```
let retrait_tous_erreurs_bis l elt =  
  let rec aux dejavu = fonction  
  | [] -> if not dejavu then failwith "Pas dedans" else []  
  | a::q -> if a = elt then aux true q else a::(aux dejavu q)  
  in aux false l;;
```