

Chapitre 2

Tableaux

1 Tableaux.

Définition simple d'un tableau :

```
let tab_1 = [| 5;7;9;4;3 |] ;;
val tab_1 : int array = [|5; 7; 9; 4; 3|]
  tab_1.(2);;  (* acces a la valeur *)
- : int = 9
tab_1.(2) <- 8;; (* changement de la valeur en position 2 *)
- : unit = ()
tab_1 ;;
- : int array = [|5; 7; 8; 4; 3|]
```

Remarque :

- Le premier élément a pour indice 0.
- Pas de constructeur de type.
- Taille fixe.

```
Array.length;;  (* pour la taille du tableau *)
- : 'a array -> int = <fun>
Array.make;;  (* creer un tableau avec la taille et un element *)
- : int -> 'a -> 'a array = <fun>
Array.make_matrix;; (* pour creer une matrice *)
- : int -> int -> 'a -> 'a array array = <fun>
```

Exercice 1: Écrire la fonction `sub_tab` qui renvoie le sous tableau de l'indice i à j inclus.

```
sub_tab : 'a array -> int -> int -> 'a array = <fun>
sub_tab [| 5;1;4;6;7;8;4;9;5|] 3 7;;
- : int array = [|6; 7; 8; 4; 9|]
```

2 Chaines de caractères.

```
let mot = "abracadabra";;
val mot : string = "abracadabra"
mot.[2];;  (* le caractere a l'indice 2 *)
- : char = 'r'
mot.[5] <- 'e';; (* pour changer un caractere *)
- : unit = ()
let salut =
  let b = "bonjour" and v = "vous" in
  (b ^ "_" ^ v);;  (* permet la concatenation *) ;;
val salut : string = "bonjour_vous"
```

Remarque :

- Le premier élément a pour indice 0.
- Pas de constructeur de type.
- Taille fixe.

```
String.length;; (* pour la taille du tableau *)  
- : 'a array -> int = <fun>  
String.make;; *( pour creer une chaine *)  
- : int -> char -> string = <fun>
```

Exercice 2: Ecrire la fonction `compte_caract` qui prend en argument une chaîne et un caractère et renvoie le nombre d'occurrence de ce caractère dans le mot.

```
val compte_caract : string -> char -> int = <fun>  
compte_caract "abracadabra" 'a';;  
- : int = 5
```