

**Question 1 Programmation fonctionnelle en Scheme**

Voici une liste de noms :

```
(define names
  '("marie" "jean" "claire" "emma" "sam" "tom" "eve" "bob"))
```

On vous demande de produire une fonction qui permettra de tirer au hasard N noms dans cette liste.

Par exemple :

```
> (winner names 3)
("claire" "jean" "eve")
> (winner names 3)
("tom" "sam" "jean")
> (winner names 2)
("eve" "jean")
```

Cette fonction utilise la fonction `random` permettant de générer des entiers aléatoires.

```
> (random 8) ;genere un entier entre 0 et 8 exclusivement
5
> (random 8)
4
> (random 8)
6
```

Pour réaliser la fonction `winner` vous devez procéder en trois étapes. Chacune de ces étapes sont indépendantes donc même si vous n'avez pas la réponse à une question vous pouvez faire la suivante.

Mais attention : vous ne devez pas utiliser la fonction `set` !

a) Créer une fonction permettant d'insérer un élément dans une liste à une position donnée

```
> (insertAt 'a '(b d e g) 2)
(b d a e g)
> (insertAt 'a '(b d e g) 1)
(b a d e g)
> (insertAt 'a '(b d e g) 20)
(b d e g a)
```

---

```
> (insertAt 'a ' (b d e g) 0)
(a b d e g)
```

- b) Pour le tirage au sort, la liste originale est mélangée en utilisant la fonction `insertAt`. C'est-à-dire en insérant le premier élément de la liste dans le reste de la liste à une position aléatoire

```
> (insertAt (car names) ; insere le premier element
      (cdr names) ; dans le reste de la liste
      (random (- (length names) 1))) ; a une position aleatoire
("jean" "claudio" "emma" "sam" "marie" "tom" "eve" "bob")
```

On applique la même fonction à cette nouvelle liste. Et on répète plusieurs fois de façon à avoir les éléments dans un ordre complètement aléatoire.

Compléter la fonction `shuffle` qui permet d'appeler récursivement `n` fois la fonction `insertAt`

```
(define (shuffle lst n)
  (if (= n 0) lst
      (shuffle ...)))

> (shuffle names 20)
("tom" "jean" "eve" "claudio" "sam" "marie" "emma" "bob")
```

- c) Afin d'obtenir les `N` personnes au hasard, il suffit maintenant d'extraire les `N` premiers noms dans la liste mélangée. Écrire la fonction `first` qui retourne la liste des `N` premiers éléments d'une liste.

```
> (first 3
      ("tom" "jean" "eve" "claudio" "sam" "marie" "emma" "bob"))
("tom" "jean" "eve")

> (first 5
      ("tom" "jean" "eve" "claudio" "sam" "marie" "emma" "bob"))
("tom" "jean" "eve" "claudio" "sam")
```

Ainsi la fonction `winner` pourra s'écrire :

```
(define (winner lst n)
  (first n (shuffle lst 20)))

> (winner names 3)
("sam" "tom" "emma")
```