

## TP n° 7

### Exercice 1

Quelle sera la valeur de  $a$  à l'issue de l'exécution des instructions suivantes ? Vérifiez.

```
> a:=3:
> f:=proc()
  local a, i:
  a:=0;
  for i from 1 to 5 do
  a:=a+i:
  od:
  RETURN(a):
end;
> f(); # appel de la fonction f
```

### Exercice 2 : Fonctions sur les listes

**Question 2.1** Ecrire une fonction `nbOcc` qui retourne le nombre d'occurrences d'un élément  $x$  dans une liste  $L$ .

**Question 2.2** Etant donnée une liste d'entiers  $L$ , écrire une fonction `maxL` qui retourne le nombre le plus grand dans  $L$ .

**Question 2.3** Ecrire une fonction qui, étant données deux listes  $L1$  et  $L2$ , retourne la valeur vraie si  $L1$  est incluse dans  $L2$  (i.e. si tous les éléments de la liste  $L1$  sont dans la liste  $L2$ ).

### Exercice 3 : Fonctions irem et iquo

Les fonctions `irem` et `iquo` calculent le reste et le quotient (respectivement) de la division entière d'un nombre  $a$  par un nombre  $b$ . Ainsi `irem(22,5)` retourne 2 et `iquo(22,5)` retourne 4.

**Question 3.1** Ecrire une fonction `nbDiv` qui retourne le nombre de diviseurs d'un entier  $n$ . Par exemple, les diviseurs de 6 étant 1, 2, 3 et 6, `nbDiv(6)` doit retourner 4.

**Question 3.2** Ecrire une fonction `SOMCHIFFR` qui calcule la somme des chiffres d'un entier quelconque  $n$  (par exemple, si l'entier  $n$  est 12304, la somme de ses chiffres est 10).

### Exercice 4 : Palindrome

Un *palindrome* est une chaîne de caractères qui se lit de la même manière dans les deux sens (par exemple "kayak"). Ecrire une fonction qui teste si une chaîne de caractères est un palindrome.

### Exercice 5 : Fonctions modifiant les listes

**Question 5.1** Ecrire une fonction `supp` qui, étant donné une liste  $L$  et un nombre  $x$ , supprime toutes les occurrences de  $x$  dans  $L$  (i.e. qui renvoie une liste formée à partir de la liste initiale en supprimant les occurrences de  $x$ ).

**Question 5.2** Ecrire une fonction `INSERT` qui insère un élément  $x$  dans une liste composée d'entiers triés dans l'ordre croissant  $L$ , en préservant le fait que  $L$  est trié.

**Exercice 6 : Fonction *sort***

Dans cet exercice, nous voulons créer notre propre fonction *sort* (qui trie les éléments d'une liste donnée en paramètre).

**Question 6.1** Ecrire une fonction **IndiceMin** qui renvoie l'indice du plus petit élément d'une liste  $L$ .

**Question 6.2** Ecrire une fonction qui, étant donnée une liste  $L$  et un nombre  $x$ , renvoie la liste où l'on a ajouté  $x$  à la fin de  $L$ .

**Question 6.3** En se servant des deux questions précédentes, écrire une fonction **MySort** équivalente à la fonction *sort*.