

Fonctions et récursivité

M2 AIGEME, année 2008-2009

Exercice 1

1. Ecrire une fonction non récursive qui calcule la factorielle d'un entier $n! = 1 \times 2 \times 3 \times \dots \times (n-1) \times n$:

```
Fonction fact_iteratif (n : Entier) : Entier
```

2. Ecrire une fonction récursive qui calcule la factorielle d'un entier $n! = 1 \times 2 \times 3 \times \dots \times (n-1) \times n$:

```
Fonction fact_recuratif (n : Entier) : Entier
```

Exercice 2

1. Ecrire une fonction récursive qui calcule la puissance entière d'un nombre réel x^n .

```
Fonction puissance (x : Réel; n : Entier) : Réel
```

2. Ecrire une fonction récursive qui renvoie un booléen donnant la parité d'un entier. Cette fonction de fera appel à aucune fonction auxiliaire.

```
Fonction est_pair (n : Entier) : Booléen
```

Exercice 3

Que fait la fonction récursive suivante ?

```
Fonction truc (n : Entier) : Entier
```

```
Debut
```

```
  Si (n<0)
```

```
    Alors Renvoyer(0)
```

```
  Sinon Renvoyer(truc(n-1)+1)
```

```
  FinSi
```

```
Fin
```

Exercice 4

Que fait la fonction récursive suivante ?

```
Fonction machin (n : Entier) : Entier
```

```
Debut
```

```
  Si (n<0)
```

```
    Alors Renvoyer(0)
```

```
  Sinon Renvoyer(machin(n)+1)
```

```
  FinSi
```

```
Fin
```

♠ Exercice 5

On définit la fonction suivante :

```
Fonction mystere(n : Entier) : Entier
```

```
Debut
```

```
  Si (n>0)
```

```
    Alors Renvoyer(n)
```

```
  Sinon Renvoyer(mystere(mystere(n+2)))
```

```
  FinSi
```

```
Fin
```

1. Quelle est la valeur de `mystere(n)` pour $n > 0$?
2. Calculer `mystere(-1)`, `mystere(-2)` et `mystere(-3)`.
3. En déduire la valeur de `mystere(n)` pour $n \leq 0$.

♠ Exercice 6

Comment pourrait-on trier un tableau d'entiers à l'aide d'une fonction récursive :

```
Fonction tri_recuratif (tab : Tableau d'Entiers) : Tableau d'Entiers
```

Indication : On pourra utiliser la fonction suivante :

```
Fonction sous_tableau (tab : Tableau d'Entiers; i,j : Entiers) : Tableau d'Entiers
```

qui prend en argument un tableau `tab` et deux indices `i` et `j` et qui renvoie le sous-tableau :

$$\begin{cases} [\text{tab}[i], \text{tab}[i+1], \dots, \text{tab}[j]] & \text{si } 0 \leq i < j < \text{long}(\text{tab}) \\ \text{le tableau vide } [] & \text{sinon} \end{cases}$$