# Travaux Pratiques de langage C n°2

## Cours d'Informatique de l'École Centrale d'Électronique

*—Ingénieur Première Année—*

---

Lors de cette séance, nous allons aborder plus précisément les points suivants :

- déclaration et définition d'une fonction ;
- appel par valeur.

---

▸ **Exercice 1. (factorielle d'un entier)** *La fonction factorielle est définie par les relations suivantes :*

$$n! = \begin{cases} 1 & \text{si } n = 0 \\ n \times (n-1) \times \cdots \times 3 \times 2 \times 1 & \text{si } n > 0 \end{cases}$$

*Écrire la fonction* `Factorielle` *qui prend en argument* `n` *et qui calcule la factorielle de* `n`*.*

▸ **Exercice 2. (coefficients binomiaux)** *Nous cherchons à calculer les coefficients binomiaux $\binom{n}{p}$ définis par les relations suivantes :*

$$\binom{n}{p} = \begin{cases} \dfrac{n!}{p!(n-p)!} & \text{si } 0 \leq p \leq n \\ 0 & \text{sinon} \end{cases}$$

*Écrire la fonction* `Binomial` *qui calcule le coefficient binomial $\binom{n}{p}$.*

▸ **Exercice 3.** *Nous cherchons à étudier les propriétés des coefficients binomiaux. Nous nous intéressons à la somme suivante :*

$$\sum_{i=0}^{n} \binom{n}{i} \quad .$$

*Écrire la fonction* `Sum` *qui prend en argument* `n` *et qui calcule cette somme. Essayer cette fonction pour $n = 2, 3, 4, 5, \ldots$. Quelle propriété déduire ?*

▸ **Exercice 4.** *Écrire une fonction* `SommeEntier` *qui prend en argument un entier et qui calcule la somme des chiffres composant cet entier.*

▸ **Exercice 5.** *Écrire une fonction* `PremierCarre` *qui prend en argument un entier* `n` *et qui calcule le premier carré supérieur à* `n`*.*

▸ **Exercice 6.** *Un nombre est dit parfait s'il est égal à la somme de ses diviseurs propres. Par exemple, 6 est parfait* ($6 = 1 + 2 + 3$)*. Ecrire une fonction* **`EstParfait`** *qui prend en argument un entier* `n` *et qui retourne 1 si* `n` *est parfait et 0 sinon. En déduire une fonction* **`Parfait`** *qui affiche les nombres parfaits inférieurs à 1000.*

▸ **Exercice 7.** *On cherche à afficher la décomposition en base 2 d'un entier* **`n`**. *Le principe est très simple. On effectue la division entière de* `n` *par 2. On trouve un quotient* $q_0$ *et un reste* $r_0$. *On recommence avec* $q_0$. *On construit ainsi une suite de restes* $(r_0, r_1, r_2, \ldots)$. *On arrête le processus dès que l'on trouve un quotient* $q_k$ *égal à 0. La décomposition de* `n` *en base 2 est alors* $(r_k, r_{k-1}, \ldots, r_2, r_1, r_0)$. *Par exemple pour* $n = 10$ *:*

$$\begin{array}{ccccccc} 10 & = & 5 \times 2 & + & 0 & \longrightarrow & r_0 = 0 \\ 5 & = & 2 \times 2 & + & 1 & \longrightarrow & r_1 = 1 \\ 2 & = & 1 \times 2 & + & 0 & \longrightarrow & r_2 = 0 \\ 1 & = & 0 \times 2 & + & 1 & \longrightarrow & r_3 = 1 \end{array}$$

*La représentation de 10 en base 2 est donc 1010.*

1. *déduire une méthode pour passer d'une écriture en base 10 à une écriture en base 2 ;*
2. *écrire un programme qui lit un entier entré au clavier et qui affiche cette décomposition. Pour l'exemple donné, on voudrait voir l'affichage suivant :*

```
1*2^3 + 0*2^2 + 1*2^1 + 0*2^0
```

▸ **Exercice 8. (nombres de Fibonacci)** *La suite de Fibonacci* $(\texttt{Fib})_n$ *est définie par* $\texttt{Fib}_0 = 0, \texttt{Fib}_1 = 1$ *et* $\texttt{Fib}_n := \texttt{Fib}_{n-1} + \texttt{Fib}_{n-2}$ *si* $n > 1$ *:*

1. *donner un procédé itératif de calcul du* $i^{\mathrm{e}}$ *terme de la suite de Fibonacci ;*
2. *en déduire la fonction* `Fib`*, calculant puis retournant* $\texttt{Fib}_{\texttt{i}}$ *;*
3. *une des nombreuses propriétés des nombres de Fibonacci est que le quotient de deux termes consécutifs converge vers le nombre d'or* $\dfrac{1+\sqrt{5}}{2}$. *Modifier la fonction précédente afin de créer une fonction de nom* `FibQuotient` *calculant une approximation du nombre d'or. La fonction prendra en argument un entier* `p` *et calculera le nombre d'or avec une précision de* $10^{-\texttt{p}}$.
4. *selon que l'on utilise le type* `float` *ou le type* `double` *pour stocker l'approximation du nombre d'or, calculer le plus petit entier* `k` *tel que :*

$$\frac{\texttt{Fib}_{k+1}}{\texttt{Fib}_k} - \frac{\texttt{Fib}_k}{\texttt{Fib}_{k-1}} = 0 \quad .$$