

Manuel de programmation de la bibliothèque **ESLib**

Présentation liminaire

La bibliothèque *ESLib* a été conçue dans le but de mettre à disposition des étudiants, dès la première année du cycle ISEN, des moyens permettant de gérer les actions courantes d'interaction avec l'utilisateur, et ce sans entrer dans la complexité de mise en œuvre du langage C.

Ainsi *ESLib*, bien qu'étant basée sur les entrées/sorties standard, permet à des débutants en langage C de demander à l'utilisateur des entrées de données, et permet d'afficher en mode texte nombre de données, avec un formalisme très simple.

Fonctionnement interne

ESLib s'attache à respecter la plus grande symétrie possible entre les entrées et les sorties de données. Ainsi il est possible de lire au clavier les données suivantes :

- un entier par *int lisEntier()* ;
- un nombre en virgule flottante par *float lisFlottant()* ;
- un caractère par *char lisCaractere()*.

Les opérations d'écriture correspondantes sont :

- *ecrisEntier(int valeur)* ;
- *ecrisFlottant(float valeur)* (le nombre de chiffres à afficher après la virgule peut être modifié à l'aide de *fixePrecision(int n)*) ;
- *ecrisCaractere(char valeur)*.

*ecrisChaine(char *chaine)* peut aussi être utilisée pour afficher une chaîne de caractères complète. *tabulation()* et *sautDeLigne()* insèrent respectivement une tabulation et un saut de ligne dans le flot courant des données.

Ces opérations sont basiques et incomplètes, car elles ne doivent être utilisées que par commodité durant les premiers temps de la formation au langage C.

Pour aller plus loin

effaceEcran(), pour des raisons de compatibilité entre systèmes du code source de la bibliothèque, affiche une série de sauts de ligne, faisant croire à un effacement de la fenêtre texte.

double tempsCPU() renvoie le nombre de seconde écoulé relativement au programme qui a été lancé.

float valeurAleatoire() renvoie une valeur aléatoire comprise entre 0 inclus et 1 exclu. Le générateur pseudo-aléatoire peut être initialisé à l'aide de *initialiseValeurAleatoire(float valeur)*.