

## TD2 - Algorithmique et C de base - Boucle WHILE

### Objectifs

Conception de programmes simples, utilisation des boucles «tant que». Premières fonctions et actions.

### 1 Analyse/Conception

On ne traduira pas les algorithmes en langage C.

**EXERCICE 1** *On suppose qu'on dispose d'une fonction `nb_au_hasard` de signature `Entier × Entiers → Entier`. L'appel `nb_au_hasard(nmin, nmax)` retourne un entier tiré au hasard entre `nmin` et `nmax` compris.*

*Concevoir un programme qui :*

- Choisit au hasard un nombre  $x$  compris entre 0 et 64.
- Demande à l'utilisateur un nombre  $y$ , dit "gagné" si  $x = y$ , et dans le cas contraire donne une indication "plus petit" ou "plus grand".
- Lorsque le nombre a été deviné, imprime le nombre de tentatives.

**A partir de maintenant, nos algorithmes seront des fonctions ou des actions**

**EXERCICE 2** *Écrire un algorithme, qui étant donnés `nbH`, `nbM`, `nbS` dénotant respectivement un nombre d'heures, minutes, secondes, décompte cette durée (en affichant tous les couples  $(h, m, s)$ ) et affiche "BOUM" en fin de décompte.*

**EXERCICE 3** *Division euclidienne :*

- Comment calculer le quotient et le reste de la division euclidienne de  $a$  par  $b$ , sans utiliser `div` ni `mod`? Illustrer avec  $a = 32$  et  $b = 7$ ?
- Et si vous n'avez pas le droit d'utiliser la multiplication?
- Écrire l'algorithme (sans multiplication) en pseudo langage,  $a$  et  $b$  sont des données (avec  $a \geq b$ ); on demande d'imprimer les résultats sur la sortie standard.
- Pourquoi ne peut-on pas utiliser de fonction?

**EXERCICE 4** *Soit  $u_n$  définie par  $u_0 = 1515$  et  $u_{n+1} = 3u_n + 42$  pour  $n \in \mathbb{N}^*$ .*

- Écrire une fonction impérative `calculeU` qui étant donné  $k$ , calcule  $u_k$ .
- Écrire la version récursive de cet algorithme.
- Écrire un algorithme qui calcule le premier  $k$  tel que  $u_k$  est strictement supérieur à 1 million.

**EXERCICE 5** (Syracuse) *Soit la suite définie par  $u_0 = N$  (entier positif) et  $u_{n+1} = \frac{u_n}{2}$  si  $u_n$  est pair et  $u_{n+1} = 3u_n + 1$  sinon. La conjecture de Syracuse dit que cette suite passe par 1. Écrire un algorithme qui vérifie cette conjecture pour tous les  $N$  de `min` à `max`.*

EXERCICE 6 *Concevoir un algorithme qui calcule et affiche le plus petit entier  $n$  tel que*  
$$\sum_{i=1}^n \left( \sum_{j=1}^i (i+j) \right) \geq 1000.$$

## 2 Fonctions C

On écrira directement le code C :

EXERCICE 7 *Soit la suite  $u$  suivante :  $u_n = \begin{cases} A & \text{si } n = 0, \\ 2 + 3 * u_{n-1} & \text{sinon.} \end{cases}$ .*

*Écrire un programme complet qui imprime les 10 premiers termes de la suite pour  $A = 3$ .*

EXERCICE 8 *Écrire une fonction `int sommation(int n)` qui étant donné un entier  $n > 0$  renvoie  $\sum_{i=1}^n i$  (la somme des  $n$  premiers entiers).*