Travaux Dirigés d'algorithmique n°3

Cours d'Informatique de Première Année

—DEUG Sciences S2-SM/STPI—

Itérations

Dans cette séance, nous abordons les différentes formes d'itérations.

Les boucles permettent de répéter plusieurs fois une même suite d'instructions. Il existe trois boucles différentes, chacune ayant sa particularité :

- la boucle `for` ;
- la boucle `while` ;
- la boucle `do ... while` ;

▸ **Exercice 1.** ***(Carrés des n premiers entiers)***
Écrire un programme qui lit un entier n et qui affiche les carrés des entiers de 1 à n. Utilisez tout d'abord le type de boucle qui semble le plus adapté, puis essayez les deux autres.

▸ **Exercice 2.** ***(Positivité d'un entier)***
Écrire un programme qui lit un entier `N` et qui :

- *affiche `N n'est pas positif, recommencez`, et attend un autre entier, tant que `N` est négatif ;*
- *affiche `N est positif, ok`, si `N` est positif.*

▸ **Exercice 3.** ***(Affichage d'un triangle)***
Écrire un programme qui lit un entier n, et qui affiche un triangle isocèle de base $2n-1$ et de hauteur n. Par exemple, pour $n = 4$:

```
*
**
***
****
***
**
*
```

Comment permettre à l'utilisateur de choisir le symbole utilisé ?

▸ **Exercice 4.**

Écrire un programme qui décompose un entier `N` *sous la forme* x^y $(x, y \in \mathbb{N})$. *Dans un premier temps, on posera* $x = 3$. *L'affichage ressemblera, selon le cas, à :*

- `N` *n'est pas une puissance de 3 ;*
- `N` *est égal à 3 puissance* `y` *(où* `y` *est la puissance calculée).*

Modifier le programme pour que l'utilisateur puisse choisir un autre nombre que 3.

▸ **Exercice 5.** ***(Recherche d'un nombre caché)***

Écrire un programme qui demande à l'utilisateur de trouver un nombre entier secret compris entre 0 *et* 1000. *Pour cela, il sera guidé par le programme qui indique à chaque essai si la valeur saisie est supérieure ou inférieure au nombre secret. L'utilisateur doit trouver le nombre secret en au plus dix essais.*
Pour le moment, vous entrerez vous même la valeur du nombre secret, vous verrez plus tard comment obtenir un nombre aléatoire.
Lorsque le jeu se termine, le programme affiche le résultat (`gagné` *ou* `perdu`*) et, en cas de gain, le nombre d'essais.*