**R101 – TD2**

## Exercice 1 : afficher 10 fois "Bonjour"

Ecrire un algorithme qui affiche 10 fois la chaine de caractères "Bonjour" en utilisant les 5 types de boucle suivants :

* pour
* jusqua
* tant\_que
* repeter ... jusqua
* boucle ... fboucle

## Exercice 2 : afficher les lignes lues

Ecrire un algorithme qui répète l'exercice 2 du TD1 (programme echo) jusqu'à ce que l'utilisateur saisisse une chaine vide (chaine de taille nulle).

« Ecrire un algorithme qui :

* affiche l'invite : "saisir une string" ;
* lit une chaine de caractères au clavier ;
* l'affiche à l'écran.

PS: les deux derniers items de ce petit programme correspondent à la commande "echo" de bash. »

## Exercice 3 : évaluer un polynôme de degré N

Le but de cet exercice est d'évaluer un polynôme de réels de degré N (de la forme a0 + a1X + a2X2 + ... aNXN). Pour cela :

* saisir N ;
* déclarer un tableau de N + 1 réels ;
* remplir ce tableau avec des valeurs saisies au clavier ;
* saisir X ;
* évaluer le polynôme ;
* afficher le résultat de l'évaluation.

Indications de mise en œuvre :

1. Vous disposez de la fonction Puissance () de signature :

Fonction Puissance (X : in reel, N : in entier\_naturel) renvoie reel ;

1. Réécrire l’algorithme sans utiliser la fonction puissance.

## Exercice 4 : Alphabet - 1

Ecrire un programme qui affiche toutes les minuscules, à raison d’une lettre par ligne.

## Exercice 5 : Alphabet - 2

Ecrire un programme qui affiche toutes les majuscules, à raison d’une lettre par ligne.

## Exercice 6 : Alphabet - 3

Ecrire un programme qui affiche tous les couples minuscules / majuscules, à raison d’un couple de lettres par ligne.

## Exercice 7 : Alphabet - 4

Ecrire un programme qui affiche tous les couples minuscules / majuscules en partant de la fin de l’alphabet.

## Exercice 8 : Dessiner une figure

Vous devez dessiner (afficher à l’écran un certain nombre de caractères) chacune des figures géométriques ci-dessous selon les spécifications suivantes :

* Un algorithme par figure ;
* Chaque figure devra être dessinée pleine et vide (*ie.* uniquement le contour) ;
* Vous devez demander à l’utilisateur de saisir chaque variable qui vous semble utile.

Les figures sont les suivantes :

* Carré
* Rectangle
* Triangle rectangle isocèle inférieur gauche ;
* Triangle rectangle isocèle inférieur droit.

Indications de mise en œuvre : pour chaque figure, vous devez écrire 2 algorithmes. L’un utilisera le type string, l’autre pas. Vous êtes libre de choisir le caractère d’affichage.