TP3 : Les structures répétitives

Exercice 1 :

Écrire un programme qui permet de lire **N** notes (N donné), puis retourne le nombre de notes qui sont supérieures à 10. La note est comprise entre 0 et 20.

Exercice 2 : Puissance

1/ Ecrire un programme Pascal qui permet de calculer A^x. ($x > 0$)

2/ Modifier le même programme pour calculer A^x quelque soit $x \in Z$.

Supplément :

3/ Calculer $(A^x + B^y) / A^x$

$(A,B) \in \Re$, $(X, Y) \in Z$, $X >= 0$ et $Y >= 0$

Exercice 3 :

Écrire un programme qui permet de lire un nombre entier et de retourner le nombre de chiffres qui le composent, puis afficher la somme de ses chiffres.

Exemple : N = 123456 composé de 6 chiffres, la somme est 21.

Exercice 4 :

Écrire un programme qui permet de lire successivement des nombres entiers et retourne ensuite la plus grande valeur ainsi que sa position dans l'ordre d'insertion. La saisie des nombres s'arrête lorsque l'utilisateur entre un zéro (0).

Exercice 5 : Tables de multiplication

1/ Ecrire un programme qui afficher la table de multiplication de n'importe quel nombre N lu à partir du clavier. (N>0)

3/ Réécrire le programme avec la boucle FOR.

4/ Afficher les tables de multiplication des nombres de 1 à 10.

Exercice 6 (facultatif) :

Écrire un programme qui permet de lire un nombre entier et d'inverser l'ordre de ses chiffres.
Exemple : N = 123456 l'inverse est 654321.

Exercice 4 : Phrase (facultatif)

Soit une phrase qui se termine par un point. Les mots de cette phrase sont séparés par un ou plusieurs espaces. Ecrire un algorithme qui permet de :

- Déterminer le nombre de mots de cette phrase.
- Afficher le mot le plus long de la phrase.

Rappel : Les structures de contrôle

WHILE - DO (tant que - faire)

```
while < Condition> do
        begin
        < Instruction / bloc d'instructions >
        end;
```

REPEAT - UNTIL (répéter - jusqu'à ce que)

```
repeat
        < Instruction / Bloc d'instructions >
until   <Condition>
```

FOR - DO (pour - faire)

```
for i:=<valeur initiale> to <valeur finale> do
begin
< Instruction / Bloc d'instructions >
end
```