

SEQUENCE 4 – STRUCTURES REPETITIVES
SEANCE 2 – ALGO DE CUMUL, MOYENNE, MIN ET MAX

OBJECTIFS

- Maîtriser les algos de bases de cumul, moyenne, min, max
- Améliorez les vérifications de saisies avec les expressions régulières

Exo 1 : COMPTER, CUMULER, COMPARER

1. Au sein du répertoire « **Sequence_4_StructuresRepetitives** », créez un projet « **Exo1_AlgoCumulMoyenne** » type console dans une solution, nommée « **Seance2_AlgoCumulMaxMin** ».

REMARQUE : PAS DE TABLEAUX !

2. Votre programme doit:

- Faire la saisie de notes pour un professeur, sachant que celui-ci ne peut prédire à l'avance le nombre de notes qu'il devra saisir.
Rem : vous pouvez utiliser le mot clef **break** pour sortir de la boucle
- Afficher le nombre de notes totales, la note maximale et la note minimale, le nombre de notes inférieures à 10, le nombre de notes supérieures ou égales à 10, la moyenne générale.

```
-----
SAISIE
-----
Note 1 ? (Q pour Quitter)
12
Note 2 ? (Q pour Quitter)
13
Note 3 ? (Q pour Quitter)
20
Note 4 ? (Q pour Quitter)
5
Note 5 ? (Q pour Quitter)
Q
-----
STATISTIQUES
-----
Nb notes : 4
Note max : 20
Note min : 5
Nb note < 10 : 1
Nb note >= 10 : 3
Moyenne générale : 12,5
```

2. Votre programme doit s'assurer des saisies : le note doit être comprise entre 0 et 20 ou Q et mettez Q, 0 et 20 en constantes !

```
-----
SAISIE
-----
Note 1 ? (Q pour Quitter)
aa
Erreur de saisie. Nombre entre 0 et 20 attendu. Recommence :
22
Erreur de saisie. Nombre entre 0 et 20 attendu. Recommence :
10
Note 2 ? (Q pour Quitter)
```

EXO 2 : TABLES DE MULTIPLICATION

1. Ajoutez un projet « **Exo2_UneTable** ». Votre programme doit demander à l'utilisateur :
 - la table qu'il désire réciter.
 - lui faire réciter la table
 - lui préciser l'exactitude ou non de chacune de ses réponses
 - lui afficher sa note totale :

Note : 8/10

Entrez la table:

3

Combien font 3*1 ?

3

Resultat correct

Combien font 3*2 ?

515

Resultat incorrect

Combien font 3*3 ?

9

2. Avez-vous utilisé des constantes pour indiquer qu'on commence à 1 et qu'on finit à 10 ? Sinon faites-le et modifiez votre code :

```
public static readonly int MIN = 1;
public static readonly int MAX = 10;
```

3. Créez un projet « **Exo2_ToutesLesTables** ». Votre programme doit faire réciter toutes les tables à l'utilisateur. Attention, pour ne pas perdre trop de temps à tester, vous modifierez la constante MAX : passez de 10 à 3. Et affichez le score : le score total possible doit être calculé à partir des constantes.

3. Créez un projet « **Exo2_ToutesLesTables_AvecMethodes** ». Tout d'abord, rendez vos constantes **globales** : placez-les au-dessus du main : ainsi vous pourrez les utiliser dans le main mais aussi dans les méthodes que vous allez définir

```
class Program
{
    public static readonly int MIN = 1;
    public static readonly int MAX = 3;
    static void Main(string[] args)
    {
```

Puis définissez les 2 méthodes suivantes, testez les l'une après l'autre :

- `public static int ReciteUneTable (uint table)` : elle reprend le code de la V1 : elle doit faire réciter la table passée en paramètre et renvoie le score.
- `public static int ReciteToutesLesTables()` : elle doit utiliser `ReciteUneTable` pour faire réciter toutes les tables et renvoie le score.

POUR LES PLUS RAPIDES

Ecrivez les programmes permettant de réaliser les affichages ci-dessous. Utilisez la structure for. Ces affichages doivent être dynamiques, ils dépendent du chiffre saisi par l'utilisateur !
Remarque : l'utilisateur peut choisir librement le chiffre, il n'est pas borné aux valeurs utilisées pour les exemples ! C'est à vous de trouver la règle qui permet de généraliser les exemples. Afin de pouvoir tester votre programme plusieurs fois de suite, faites une boucle pour éviter ainsi d'avoir à le relancer à chaque fois !

Affichage 1 : (4 étoiles par ligne

si l'utilisateur saisit 10

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

si l'utilisateur saisit 5

```
****
*
```

si l'utilisateur saisit 2

```
**
```

Affichage 2 :

si l'utilisateur saisit 3

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

si l'utilisateur saisit 6

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

si l'utilisateur saisit 2

```
*
**
```

Affichage 3 :

si l'utilisateur saisit 3

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

si l'utilisateur saisit 6

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

si l'utilisateur saisit 2

```
**
*
```

Affichage 4 :

si l'utilisateur saisit 3

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

si l'utilisateur saisit 5

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

si l'utilisateur saisit 2

```
**
*
```