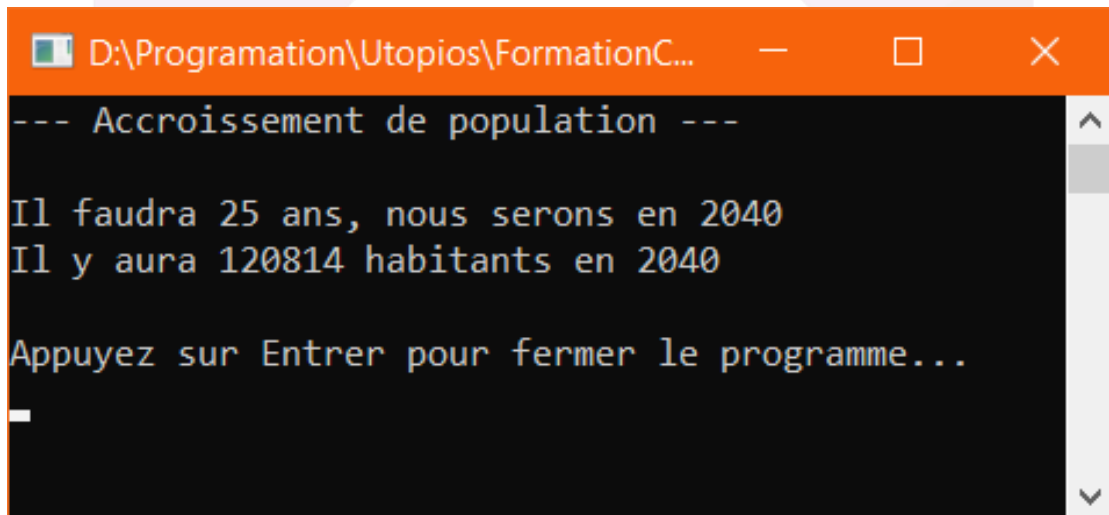


Exercice 23 – Accroissement de la population

Objet : Comprendre les fonctionnalités des structures conditionnelles **For** imbriquées avec des variables **int**.

Sujet : Créez un nouveau projet dans la solution « **ExercicesCSharp** » nommé « **Exercice23** » et faire un programme permettant de connaître le nombre d'années nécessaire à une population pour atteindre un certain nombre de personnes.

- L'accroissement de la population de Tourcoing est de 0,89%
- En 2015, la ville comptait 96809 habitants
- Combien d'années faut-il pour atteindre 120.000 habitants ?
- Combien y aura-t-il d'habitants cette année-là ?



```
--- Accroissement de population ---  
  
Il faudra 25 ans, nous serons en 2040  
Il y aura 120814 habitants en 2040  
  
Appuyez sur Entrer pour fermer le programme...  
_
```

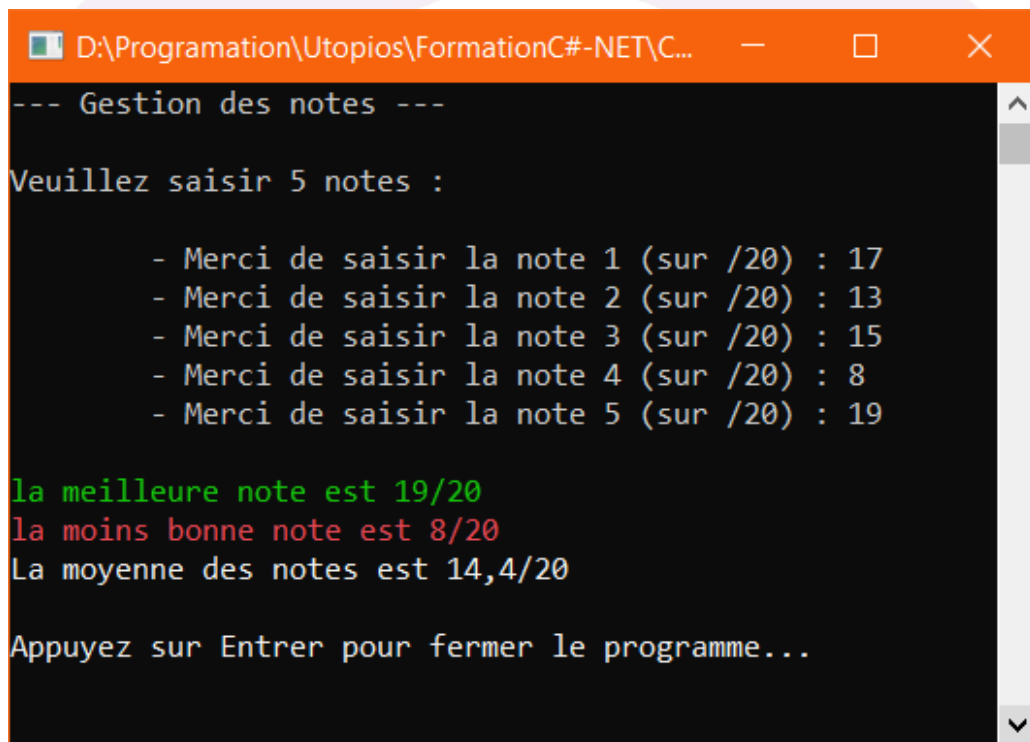
- Effectuer l'affichage du résultat dans la console comme indiqué dans l'exemple ci-dessus.

Exercice 25 – Gestion des notes

Objet : Comprendre les fonctionnalités des structures conditionnelles **For** avec des variables **int** et **double**.

Sujet : Créez un nouveau projet dans la solution « **ExercicesCSharp** » nommé « **Exercice25** » et faire un programme permettant de connaître, pour une saisie de 5 notes :

- La meilleure note
- La moins bonne note
- La moyenne des notes



```
--- Gestion des notes ---  
Veuillez saisir 5 notes :  
  
- Merci de saisir la note 1 (sur /20) : 17  
- Merci de saisir la note 2 (sur /20) : 13  
- Merci de saisir la note 3 (sur /20) : 15  
- Merci de saisir la note 4 (sur /20) : 8  
- Merci de saisir la note 5 (sur /20) : 19  
  
la meilleure note est 19/20  
la moins bonne note est 8/20  
La moyenne des notes est 14,4/20  
  
Appuyez sur Entrer pour fermer le programme...
```

- Effectuer l’affichage des résultats dans la console comme indiqué dans l’exemple ci-dessus.

- Écrire un programme qui permet d'afficher un triangle isocèle formé d'étoile *.
- La hauteur du triangle (le nombre de lignes) sera saisie, comme dans l'exemple ci-contre.
- Il existe plusieurs méthodes pour arriver au résultat.

Quelques pistes : f-strings, mathématiques, for imbriqués, incrémentation et décrémentation

```
saisir la hauteur du triangle : 10
```

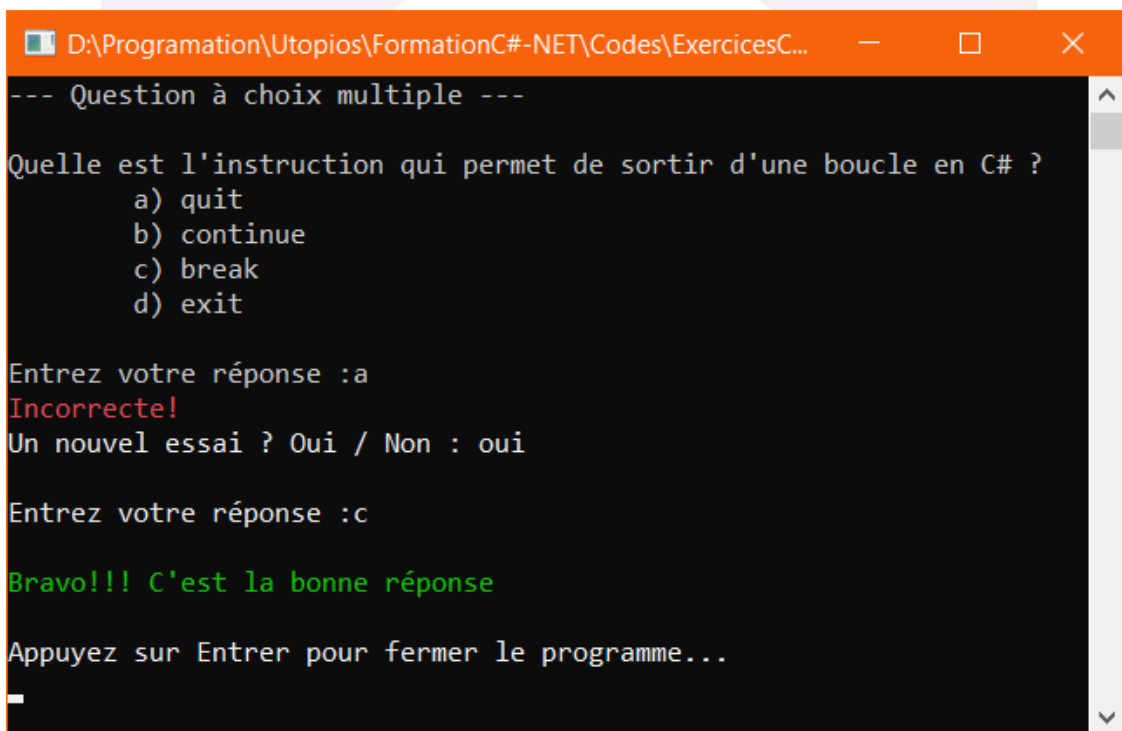
```
      *
     ***
    *****
   ********
  **********
 **********
 **********
 **********
 **********
 **********
```

Exercice 30 – Question à choix multiple

Objet : Comprendre les fonctionnalités des structures conditionnelles **do...while** avec des variables **string**.

Sujet : Créez un nouveau projet dans la solution « **ExercicesCSharp** » nommé « **Exercice30** » et faire un programme permettant de répondre à une question à choix multiple.

- La réponse doit fonctionner pour une saisie majuscule ou minuscule
- Indiquez à l'utilisateur quand la réponse est incorrecte.
- Proposez de faire un nouvel essai.
- Une réponse correcte impose la fin du programme



```
--- Question à choix multiple ---
Quelle est l'instruction qui permet de sortir d'une boucle en C# ?
  a) quit
  b) continue
  c) break
  d) exit

Entrez votre réponse :a
Incorrecte!
Un nouvel essai ? Oui / Non : oui

Entrez votre réponse :c
Bravo!!! C'est la bonne réponse

Appuyez sur Entrer pour fermer le programme...
_
```

- Effectuer l'affichage du résultat dans la console comme indiqué dans l'exemple ci-dessus.

Exercice 31 – Gestion des notes (Menu)

Objet : Comprendre les fonctionnalités des structures conditionnelles **do...while** avec des variables **double** et **int**.

Sujet : Créez un nouveau projet dans la solution « **ExercicesCSharp** » nommé « **Exercice31** » et refaire le programme permettant de connaître la meilleure, la moins bonne et la moyenne, pour une saisie de « n » notes :

- La meilleure note
- La moins bonne note
- La moyenne des notes

```
--- Gestion des notes avec menu ---
1---Saisir les notes
2---La plus grande note
3---La plus petite note
4---La moyenne des notes
0---Quitter
Faites votre choix : 1

----- Saisir les notes -----
(999 pour stoper la saisie)
Merci de saisir la note 1 (sur /20) : 13
Merci de saisir la note 2 (sur /20) : 15
Merci de saisir la note 3 (sur /20) : 22
Merci de saisir la note 3 (sur /20) : 999
Erreur de saisie, la note est sur 20 !
Merci de saisir la note 3 (sur /20) : 16
Merci de saisir la note 4 (sur /20) : 999

----- La plus grande note -----
La note la plus grande est : 16/20

--- Gestion des notes avec menu ---
1---Saisir les notes
2---La plus grande note
3---La plus petite note
4---La moyenne des notes
0---Quitter
Faites votre choix : 2

----- La plus petite note -----
La note la plus petite est : 13/20

--- Gestion des notes avec menu ---
1---Saisir les notes
2---La plus grande note
3---La plus petite note
4---La moyenne des notes
0---Quitter
Faites votre choix : 4

----- La moyenne des notes -----
La moyenne est de : 14,7/20

--- Gestion des notes avec menu ---
1---Saisir les notes
2---La plus grande note
3---La plus petite note
4---La moyenne des notes
0---Quitter
Faites votre choix : 0
```

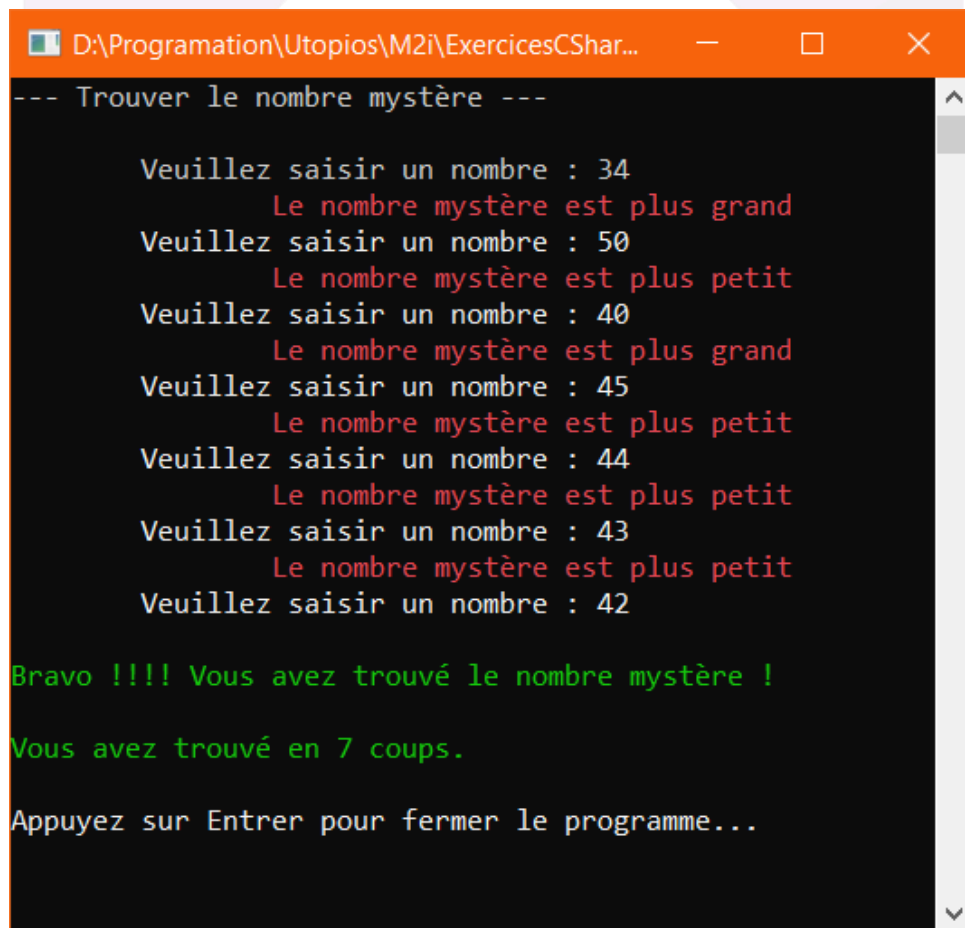
- Vous devez concevoir une interface utilisateur (User Interface) en utilisant deux **do...while** et un **switch**.
- Cette interface devra fonctionner tant que l'utilisateur ne saisit pas 0. Pour fermer le programme, utilisez la méthode : `Environment.Exit(0);`
- Afin de nettoyer la console entre chaque menu, utilisez : `Console.Clear();`
- Effectuer l'affichage des résultats dans la console comme indiqué dans les exemples ci-dessus.

Exercice 28 – Le nombre Mystère

Objet : Comprendre les fonctionnalités des structures conditionnelles **While** avec des variables **int** et **bool**.

Sujet : Créez un nouveau projet dans la solution « **ExercicesCSharp** » nommé « **Exercice28** » et faire un programme permettant de trouver un nombre généré aléatoirement entre 1 et 50.

- Indiquez à l'utilisateur que le nombre mystère est soit plus grand ou plus petit que la saisie tant que la bonne réponse n'est pas trouvée.
- Pour déclarer un nombre aléatoire : `Random aleatoire = new Random();`
- Pour générer un nombre aléatoire après déclaration : `int nbMystere = aleatoire.Next(1, 51);`
- Comptez le nombre de coups et l'afficher une fois la partie terminée.



```
--- Trouver le nombre mystère ---

Veillez saisir un nombre : 34
    Le nombre mystère est plus grand
Veillez saisir un nombre : 50
    Le nombre mystère est plus petit
Veillez saisir un nombre : 40
    Le nombre mystère est plus grand
Veillez saisir un nombre : 45
    Le nombre mystère est plus petit
Veillez saisir un nombre : 44
    Le nombre mystère est plus petit
Veillez saisir un nombre : 43
    Le nombre mystère est plus petit
Veillez saisir un nombre : 42

Bravo !!!! Vous avez trouvé le nombre mystère !

Vous avez trouvé en 7 coups.

Appuyez sur Entrer pour fermer le programme...
```

- Effectuer l'affichage du résultat dans la console comme indiqué dans l'exemple ci-dessus.

Écrivez un programme qui affiche les nombres de 1 à 100, mais en respectant les règles suivantes :

1. Si un nombre est divisible par ****3****, affichez ****Fizz**** à la place du nombre.
2. Si un nombre est divisible par ****5****, affichez ****Buzz**** à la place du nombre.
3. Si un nombre est divisible à la fois par ****3**** et ****5****, affichez ****FizzBuzz****.
4. Sinon, affichez simplement le nombre.

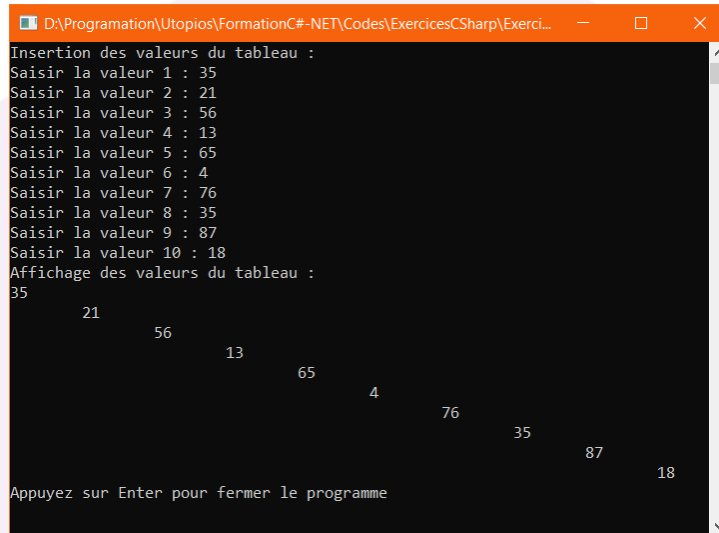
Voici un extrait du résultat attendu lorsque votre programme est exécuté :

```
```Shell
1
2
Fizz
4
Buzz
Fizz
7
8
Fizz
Buzz
11
Fizz
13
14
FizzBuzz
16
17
Fizz
19
Buzz
:::
```
```

Exercice 32 – Insertion de valeurs

Objet : Comprendre les fonctionnalités des structures conditionnelles avec des **tableaux**.

Sujet : Créez un nouveau projet dans la solution « **ExercicesCSharp** » nommé « **Exercice32** » et faire un programme permettant de le remplir :



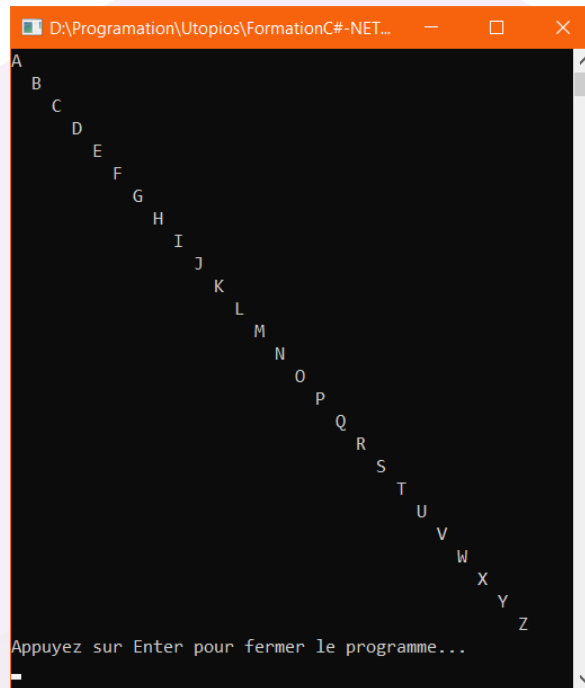
```
Insertion des valeurs du tableau :
Saisir la valeur 1 : 35
Saisir la valeur 2 : 21
Saisir la valeur 3 : 56
Saisir la valeur 4 : 13
Saisir la valeur 5 : 65
Saisir la valeur 6 : 4
Saisir la valeur 7 : 76
Saisir la valeur 8 : 35
Saisir la valeur 9 : 87
Saisir la valeur 10 : 18
Affichage des valeurs du tableau :
35
21
56
13
65
4
76
35
87
18
Appuyez sur Enter pour fermer le programme
```

- Vous devez créer un tableau permettant de contenir 10 valeurs.
- Développez l'interface afin que l'utilisateur puisse saisir les 10 valeurs afin de les stocker dans le tableau.
- Enfin procédez à l'itération de son contenu afin de l'afficher dans la console
- Effectuer l'affichage des résultats dans la console comme indiqué dans les exemples ci-dessus.
- Pour aller plus loin créez une fonction permettant de le remplir automatiquement.

Exercice 33 – La table ASCII

Objet : Comprendre les fonctionnalités des structures conditionnelles avec des **tableaux**.

Sujet : Créez un nouveau projet dans la solution « **ExercicesCSharp** » nommé « **Exercice33** » et faire un programme permettant de convertir la [table ASCII](#) :

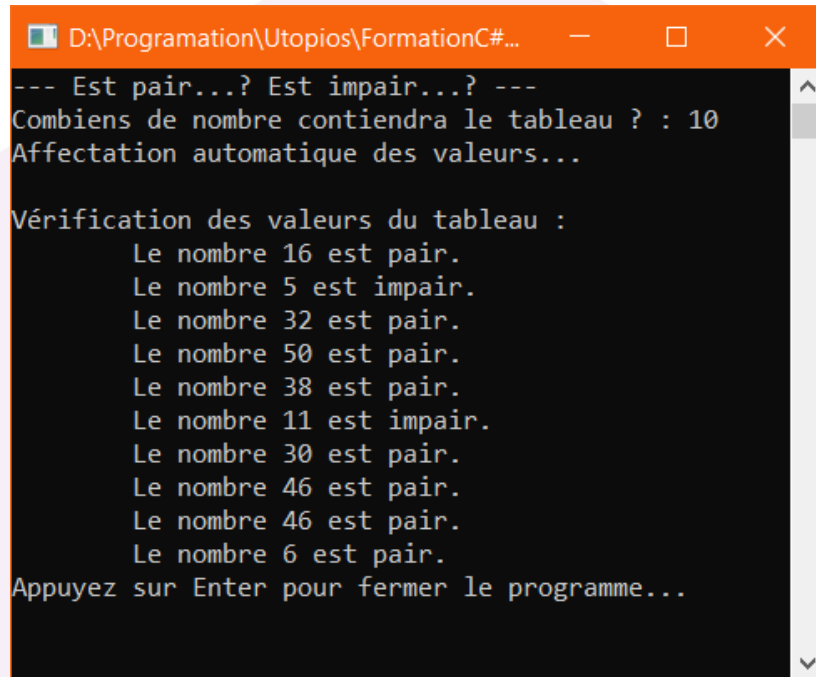


- Vous devez créer un tableau de la taille de l'alphabet.
- Trouvez une solution pour le remplir par les codes décimaux de la table ASCII afin d'obtenir l'alphabet en MAJUSCULE.
- Enfin procédez à l'itération de son contenu afin de l'afficher dans la console
- Effectuer l'affichage des résultats dans la console comme indiqué dans les exemples ci-dessus.

Exercice 34 – Est pair... ? Est impair ?

Objet : Comprendre les fonctionnalités des structures conditionnelles avec des **tableaux**.

Sujet : Créez un nouveau projet dans la solution « **ExercicesCSharp** » nommé « **Exercice34** » et faire un programme permettant de remplir automatiquement un tableau de nombre et vérifier s'ils sont pairs ou impairs :



```
D:\Programation\Utopios\FormationC#... --- □ ×
--- Est pair...? Est impair...? ---
Combiens de nombre contiendra le tableau ? : 10
Affectation automatique des valeurs...

Vérification des valeurs du tableau :
    Le nombre 16 est pair.
    Le nombre 5 est impair.
    Le nombre 32 est pair.
    Le nombre 50 est pair.
    Le nombre 38 est pair.
    Le nombre 11 est impair.
    Le nombre 30 est pair.
    Le nombre 46 est pair.
    Le nombre 46 est pair.
    Le nombre 6 est pair.
Appuyez sur Enter pour fermer le programme...
```

- Vous devez demander à l'utilisateur le nombre de chiffre que comportera le tableau
- Le programme générera aléatoirement des nombres entre 1 et 50 pour remplir les cellules du tableau.
- Enfin votre programme vérifiera s'ils sont pair ou impair lors de leur affichage
- Effectuer l'affichage des résultats dans la console comme indiqué dans les exemples ci-dessus.

Exercice 35 – Gestion des Contacts

Objet : Comprendre les fonctionnalités des structures conditionnelles **do...while**, **Switch** **Case** avec des variables **string** et **int**.

Sujet : Créez un nouveau projet dans la solution « **ExercicesCSharp** » nommé « **Exercice35** » et faire un programme permettant stocker une liste de contacts :

- Ces contacts sont stockés dans un tableau
- Au démarrage de l'application, le nombre de contacts sera demandé à l'utilisateur

```
===== Gestion des Contacts =====
Merci de saisir le nombre de contact : 5

----- Ma liste de contacts -----
1----Saisir des contacts
2----Afficher mes contacts
0----Quitter

Faites votre choix :

----- Saisir les contacts -----
Nom et prénom du contact N° 1 : Duss Jean-Claude
Nom et prénom du contact N° 2 : Delarotule Jacques
Nom et prénom du contact N° 3 : Papin Jean-Pierre
Nom et prénom du contact N° 4 : Tapis Bernard
Nom et prénom du contact N° 5 : Newton Isaac

----- Affichage des contacts -----
-Contact N° 1 : Duss Jean-Claude
-Contact N° 2 : Delarotule Jacques
-Contact N° 3 : Papin Jean-Pierre
-Contact N° 4 : Tapis Bernard
-Contact N° 5 : Newton Isaac

----- Ma liste de contacts -----
1----Saisir des contacts
2----Afficher mes contacts
0----Quitter

Faites votre choix :
```

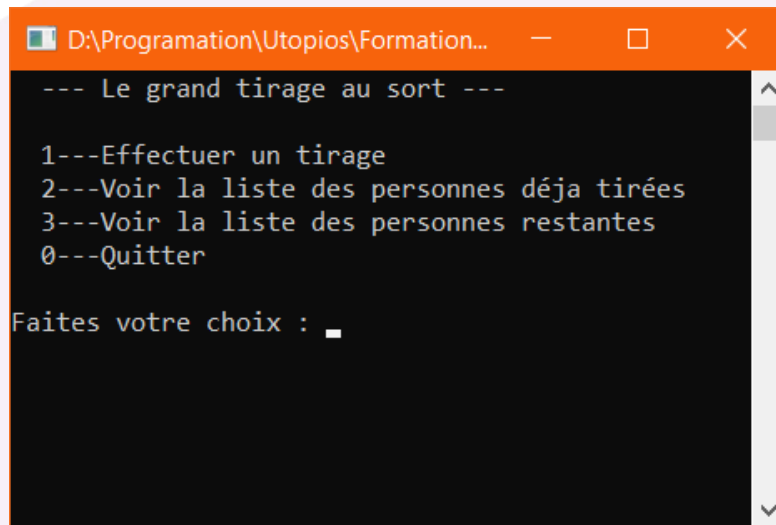
- Vous devez concevoir une interface utilisateur (User Interface) en utilisant un **do...while** et un **switch**.
- Cette interface devra fonctionner tant que l'utilisateur ne saisit pas 0. Pour fermer le programme, utilisez la méthode : `Environment.Exit(0);`
- Afin de nettoyer la console entre chaque menu, utilisez : `Console.Clear();`
- Effectuer l'affichage des résultats dans la console comme indiqué dans les exemples ci-dessus.

Exercice 39 – The Great TP !

Objet : Comprendre les fonctionnalités des structures conditionnelle **do...while**, **Switch Case** et d'énumération **foreach** avec des **tableaux**.

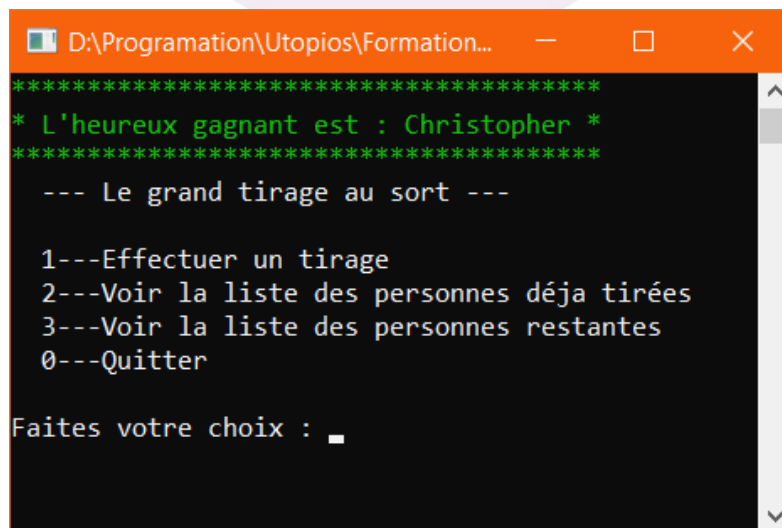
Sujet : Créez un nouveau projet dans la solution « **ExercicesCSharp** » nommé « **Exercice39** » et faire un programme permettant de tirer au sort un participant présent dans un tableau :

- Les prénoms sont stockés « en dur » dans un tableau
- Au démarrage de l'application, l'application vous propose un menu comme suit



```
D:\Programation\Utopios\Formation... --- □ ×  
--- Le grand tirage au sort ---  
  
1---Effectuer un tirage  
2---Voir la liste des personnes déjà tirées  
3---Voir la liste des personnes restantes  
0---Quitter  
  
Faites votre choix : _
```

- En tapant 1, l'application effectue un tirage au sort parmi les personnes présente dans le tableau. Chaque personne ne peut être tiré qu'une fois et son nom sera stocké dans un 2nd tableau « personne déjà tirées » et comparé à chaque tirage.



```
D:\Programation\Utopios\Formation... --- □ ×  
*****  
* L'heureux gagnant est : Christopher *  
*****  
--- Le grand tirage au sort ---  
  
1---Effectuer un tirage  
2---Voir la liste des personnes déjà tirées  
3---Voir la liste des personnes restantes  
0---Quitter  
  
Faites votre choix : _
```

- En tapant 2, on peut voir la liste des personnes figurant dans ce tableau « personne déjà tirées »

```
D:\Programation\Utopios\Formatio...
*****
* Liste des personnes déjà été tirés : *
*****
Sébastien
Allan
  Dominique
    Eric
      Anthony
        Yves
          Aguit
            Adrien
              Jérôme
                Walid
                  Olivier
                    Stéphanie
                      Yannick
                        Christopher
                          Brice

--- Le grand tirage au sort ---

1---Effectuer un tirage
2---Voir la liste des personnes déjà tirées
3---Voir la liste des personnes restantes
0---Quitter

Faites votre choix : _
```

- En tapant 3 on peut voir la liste des personnes qui n'ont pas encore été tirées

```
D:\Programation\Utopios\Formatio...
*****
* N'ont pas encore fait de correction : *
*****
Jérôme
Yves
  Aguit
    Olivier
      Walid
        Stéphanie
          Dominique
            Brice
              Yannick
                Anthony
                  Christopher
                    Allan
                      Adrien
                        Eric
                          Sébastien

--- Le grand tirage au sort ---

1---Effectuer un tirage
2---Voir la liste des personnes déjà tirées
3---Voir la liste des personnes restantes
0---Quitter

Faites votre choix :
```

- Une fois que toutes les personnes ont été tirées au sort, le tableau des personnes tirées sera remis à 0 afin de pouvoir recommencer une nouvelle série.