

6.1)

Créez une fonction qui prend un mot et détermine s'il est pluriel ou singulier. Un mot pluriel est celui qui se termine par « s ». S'il est pluriel renvoyer TRUE sinon FALSE. Vous utilisez les fonction

EndsWith() qui vérifie si une chaîne de caractère se termine par un caractère :

Exemple de la fonction EndsWith() :

```
string maChaine = "enfants";  
if (maChaine.EndsWith('s')) Console.WriteLine("pluriel");  
else Console.WriteLine("singulier");
```

Exemple:

```
check("enfants") → True  
check("filles") → True  
check("fille") → False  
check("enfant") → False
```

6.2) Créez une fonction qui prend un tableau de nombres et renvoie le plus petit nombre.

Exemple:

```
getSmall([9, 7, 1, 5]) → 1  
getSmall([100, -100, 1, 50]) → -100  
getSmall([9, 9, 9, 9]) → 9
```

6.3) Créez une fonction qui prend un tableau de nombres et renvoie le plus grand nombre du tableau.

Exemple:

```
getBigNbr([9, 7, 1, 5]) → 9  
getBigNbr([100, -100, 1, 50]) → 100  
getBigNbr([9, 9, 9, 9]) → 9
```

6.4) Écrire un programme avec les fonctions `carre` et `cube` qui calculent respectivement le carré et le cube (ou puissance 3) d'un nombre de type `double`. La méthode `main` doit tester ces deux fonctions sur plusieurs exemples.

6.5) Bien souvent, on veut comparer deux tableaux selon une égalité selon laquelle les deux tableaux `t1` et `t2` sont égaux si et seulement si ils ont la même longueur et les éléments de même indice sont égaux, c'est à dire que `t1[i] == t2[i]` pour tout indice `i`. Écrire une fonction appelée `estEgal` qui réalise ce test d'égalité pour des tableaux de type `int[]`.

6.6) Écrire une fonction qui cherche si un élément appartient à un tableau de `char`. Le caractère recherché et le tableau seront les deux paramètres de la fonction.

6.7) Écrire une fonction qui compte le nombre d'occurrences d'un caractère dans un tableau, c'est à dire le nombre de fois où un élément apparaît dans un tableau de caractères. Le caractère recherché et le tableau seront les deux paramètres de la fonction.