

Ex31** : Donner le nombre d'étapes effectuées par la fonction, en déduire la complexité en temps (Notation de Landau) de cet algorithme en C#

```
public static double sommeTableau(double[] monTableau)
{
    double somme = 0;
    for(int i = 0; i < monTableau.Length; i++)
    {
        somme += monTableau[i];
    }
    return somme;
}
```

N

$- 2 \text{ opération}$

1 opération

$$(1+1+1) \times N = O(N)$$

Ex32** : Donner le nombre d'étapes effectuées par la fonction, en déduire la complexité en temps (Notation de Landau) de cet algorithme en C#

La matrice sera considéré carré

```
public static double sommeMatrice(double[, ] maMatrice)
{
    double somme = 0;
    for (int i = 0; i < maMatrice.GetLength(0); i++) //Parcours lignes
    {
        for (int j = 0; j < maMatrice.GetLength(1); j++) //Parcours colonnes
        {
            somme += maMatrice[i, j];
        }
    }
    return somme;
}
```

N=2

2 opérations

2 opérations

$(1+1+1) \times 2$ DONC $O(n^2)$