

## Algorithmique Avancée & Complexité TD2- Complexité

### Exercice 1 :

Calculer la complexité en nombre d'opérations arithmétiques des algorithmes suivants au meilleur et au pire des cas.

### Algo 1 :

```
Pour i de 1 à n faire
  A[i] ← 0
  Pour j de 1 à n faire
    A[i] ← A[i] + B[i,j]
    D[i,j] ← 0
    Si B[i,j] > 0 alors
      Pour k de 1 à n faire
        D[i,j] ← D[i,j] + B[i,k] * C[k,j]
      Finpour
    Sinon D[i,j] ← B[i,j] + C[i,j]
  Finsi
Finpour
```

### Algo 1 :

```
Si x > 0 alors
  Pour i de 1 à n faire
    A[i] ← B[i] * C[j]
    Pour j de 1 à n faire
      D[i,j] ← 0
      Pour k de 1 à j faire
        D[i,j] ← D[i,j] + X[i,k] * Y[k,j]
      Finpour
    Finpour
  Finpour
Sinon
  Pour i de 1 à n faire
    Pour j de i à n faire
      D[i,j] ← X[i,j] + Y[i,j]
    Finpour
  Finpour
Finsi
```

**Exercice 2 :**

1. En utilisant les limites trouvez une relation en notation asymptotique entre les deux fonctions.
  - a.  $f(n) = n^3/2$  et  $g(n) = 37 n^2 + 120 n + 17$
  - b.  $f(n) = 5 n^3 + 30 n^2$  et  $g(n) = 2n^2 + 3 n^3 \log n$
2. En utilisant les propriétés de la notation Grand- **O** simplifier les expressions suivantes pour les cas  $f(n) = O(1)$ ,  $f(n) = O(n)$  et  $f(n) = O(n^2)$  :
  - a.  $O(f(n) \log f(n) + n)$
  - b.  $O(n^2 + f(n) \log f(n) + n \log n + f(n)^2)$
  - c.  $O(n^3 + 2n^2 f(n)^2 + O(n^2 + n f(n)))$

**Exercice 3 :**

Enoncer les algorithmes Tri suivants et calculer leurs complexités en termes de comparaisons :

Tri Sélection  
Tri Bulles  
Tri permutation  
Tri insertion