Université de Sidi Bel Abbès
Département d’Informatique
Master1 WIC /ISI / RSSI 2021/2022

# Algorithmique et Complexité
# TP3 : Comparaison entre 2 algorithmes pour la multiplication matricielle

Le but de ce TP est la Comparaison entre l’algorithme cubique classique et l’algorithme de Strassen pour la multiplication matricielle.
Soit 3 matrice A, B, C de taille n*n :

1- **Description de l’algorithme classique :** le calcul de C=A*B pour n=2 se fait selon la formule suivante :

$$\begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix} = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \times \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}$$

$$C = \begin{pmatrix} A_{11}*B_{11}+A_{12}*B_{21} & A_{11}*B_{12}+A_{12}*B_{22} \\ A_{21}*B_{11}+A_{22}*B_{21} & A_{21}*B_{12}+A_{22}*B_{22} \end{pmatrix}$$

## 2-Description de l’algorithme de Strassen :

L’algorithme de Strassen est un algorithme de type « diviser pour régner » dont l’objectif est de minimiser le nombre de multiplications. Le produit de deux matrices 2 × 2 peut être effectué avec seulement 7 multiplications au lieu de 8 avec la méthode classique. Cet algorithme ne s'applique que sur les matrices dont la taille est une puissance de 2. L'algorithme de Strassen est récursif : à chaque étape la matrice est divisée en quatre sous-matrices. Le cas d'arrêt de la récursivité est celui où les matrices sont de taille 1x1. Les calculs se font selon les formules suivantes :

$$\begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix} = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \times \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}$$

$$M_1 = (A_{11} + A_{22})\ (B_{11} + B_{22})$$
$$M_2 = (A_{21} + a_{22})\ B_{11}$$
$$M_3 = A_{11}\ (B_{12} - B_{22})$$
$$M_4 = A_{22}\ (B_{21} - B_{11})$$
$$M_5 = (A_{11} + a_{12})\ B_{22}$$
$$M_6 = (A_{21} - a_{11})\ (B_{11} + B_{12})$$
$$M_7 = (A_{12} - A_{22})\ (B_{21} + B_{22})$$

$$C = \begin{pmatrix} M_1 + M_4 - M_5 + M_7 & M_3 + M_5 \\ M_2 + M_4 & M_1 + M_3 - M_2 + M_6 \end{pmatrix}$$

Ici les additions et soustractions sont des additions et soustractions de matrices.

L'algorithme est le suivant :

```
int [][] Strassen(int [][] A, int [][] B, int n)
{
// n : nombre de ligne, de colonnes
// Si n n'est pas une puissance de 2 alors   on ajoute des lignes et des colonnes de
0 afin d'accéder à la puissance de 2 la plus proche supérieurement
if ( n==0)   C[0] [0]=A[0][0]*B[0][0];
 else
 {
Décomposer chacune des matrices A et B en 4 sous matrices de taille n/2 × n/2 ;
M1=Strassen(A11+A22,B11 + B22,n/2);
M2=Strassen(A21 + A22,B11 ,n/2);
M3=Strassen(A11 ,B12- B22,n/2);
M4=Strassen(A22,B21-B11,n/2);
M5=Strassen(A11+A12,B22,N/2);
M6=Strassen(A21-A11,B11+B12,n/2);
M7=Strassen(A12 - A22,B21+B22,n/2);
C11=M1 + M4 – M5 + M7;
C12=M3 + M5;
C21=M2+ M4;
C22=M1 + M3 –M2 +M6;
Composer la matrice C à partir de C11, C12, C21, C22;
Return C;
 }
```

## Questions :

1. **Lire attentivement puis compléter le code java de la classe StrassenMult, en tenant compte de toutes les indications qui y sont données.**
2. Afficher le nombre de multiplications exécutées par chaque algorithme.
3. Afficher le temps d'exécution pour chaque algorithme.
4. Faire plusieurs exécutions en modifiant la taille des matrices, en prenant des tailles avec des valeurs paires et impaires. Vérifier que les matrices C obtenues par les deux algorithmes sont les mêmes.
5. Faire des captures d'écran pour n=2 ; n=7 ; n=8.
6. Remplir le tableau suivant :

| n | | Algorithme classique | Algorithme de Strassen |
|---|---|---|---|
| | Nombre de multiplications | | |
| **2** | Temps d'exécution | | |
| **7** | Nombre de multiplications | | |
| | Temps d'exécution | | |
| **8** | Nombre de multiplications | | |
| | Temps d'exécution | | |
| **50** | Nombre de multiplications | | |
| | Temps d'exécution | | |
| **64** | Nombre de multiplications | | |
| | Temps d'exécution | | |
| | **Complexité temporelle** | | |

**Les calculs des complexités temporelles doivent être justifiés, pour les deux algorithmes.** Pour l'algorithme de Strassen, il faut donner l'équation de récurrence et la résoudre avec une méthode de votre choix.

7. Que peut –on conclure ?

**Directives du TP :**

Envoyer **deux documents** séparés, sur **classroom** **avant le jeudi 18/11/2021 à 23h 59mn. :**
1-Un document contenant le code java complet.
2- Un document contenant :
-Noms, prénoms, spécialité, groupe **du monôme ou binôme.** Tout document ne comportant pas toutes ces informations ne sera pas corrigé.
-Les captures d'écran.
- Les réponses aux questions.

**La séance de TP en présentiel sera consacrée à la consultation et à la correction du TP. Vous devez obligatoirement vous présenter avec le code que vous avez déjà développé et déjà envoyé sur classroom.**