

Chapitre 3 : Introduction à la programmation scientifique

Langages : *MATLAB / C++*

- Initiation à l'environnement **MATLAB** : scripts, fonctions, graphiques
 - Programmation de calculs géotechniques simples (ex. : calcul de tassement, contraintes)
 - Bases du langage *C++* orientées vers le calcul scientifique
 - Comparaison **MATLAB** vs *C++* : avantages et limites
- **TP proposés** : Script *MATLAB* pour la courbe de tassement - *C++* pour calcul de portance

1. Introduction

Dans les projets géotechniques, certaines analyses ou calculs spécifiques ne peuvent pas toujours être réalisés directement à l'aide d'un logiciel prêt à l'emploi. La **programmation scientifique** permet de créer des outils personnalisés pour modéliser des phénomènes, automatiser des calculs ou visualiser des données selon des besoins précis.

Ce chapitre vise à initier les étudiants à deux environnements de programmation essentiels dans l'ingénierie : **MATLAB** (orienté calcul numérique et graphique) et *C++* (langage généraliste performant). L'objectif n'est pas de former des informaticiens, mais de donner les bases nécessaires pour développer **leurs propres petits outils d'analyse géotechnique**.

2. Objectifs du chapitre

- Comprendre les principes fondamentaux de la programmation
- Maîtriser les bases de MATLAB pour les calculs numériques et les visualisations
- Introduire les fondements de C++ appliqués à des cas simples
- Savoir créer des scripts utiles en géotechnique (calcul de portance, contraintes, tassements)
- Développer l'autonomie dans la conception d'algorithmes adaptés

3. Pourquoi programmer en géotechnique ?

- Automatiser des **calculs répétitifs** (ex : coefficient de réaction du sol, module pressiométrique)
- Tester rapidement des **hypothèses géotechniques**
- Créer des **modèles personnalisés** pour les cas non couverts par les logiciels standards
- Intégrer ses propres modèles dans des workflows SIG ou BIM

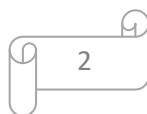
4. Initiation à MATLAB

4.1 Interface MATLAB

- Espace de travail, éditeur de scripts, fenêtre de commande
- Types de fichiers : .m (scripts et fonctions)

4.2 Bases de programmation

- Déclaration de variables, vecteurs, matrices
- Opérations mathématiques de base



- Boucles for, while, structures conditionnelles if, else

4.3 Fonctions utiles en géotechnique

- plot, semilogx, loglog : courbes sur échelles normales/log
- interp1 : interpolation linéaire (ex. entre essais)
- polyfit, polyval : régression polynomiale (ajustement de courbes expérimentales)

4.4 Exemples pratiques

- Script MATLAB pour calculer les tassements par la méthode des couches
- Modélisation de la courbe de consolidation oedométrique

5. Introduction à C++

5.1 Présentation du langage

- Langage compilé, très rapide
- Structure d'un programme C++ (fonction main, bibliothèques, syntaxe stricte)

5.2 Concepts de base

- Déclaration de variables, types (int, double, string...)
- Boucles for, while, conditions if/else
- Entrée/sortie de données (cin, cout)

5.3 Applications en géotechnique

-  Programme C++ pour calcul de la capacité portante d'une semelle
-  Création d'une table de calcul de contraintes selon la profondeur

6. Applications pratiques (TP à proposer)

TP	Langage	Objectif
TP1	MATLAB	Script de calcul de tassement sous semelle circulaire
TP2	MATLAB	Courbe de contrainte-déformation et module de déformation
TP3	C++	Programme de calcul de la contrainte admissible selon la formule de Terzaghi
TP4	C++	Générateur de tableau de profondeur vs pression effective

7. MATLAB vs C++ : synthèse comparative

Critère	MATLAB	C++
Simplicité d'utilisation	Très élevée	Moyenne
Puissance graphique	Excellente	Faible
Rapidité d'exécution	Moyenne	Très élevée
Type d'usage	Calculs scientifiques, prototypage rapide	Déploiement, haute performance

8. Conclusion

Ce chapitre a permis d'introduire les bases de la **programmation scientifique**, un atout stratégique dans la formation d'un ingénieur géotechnicien moderne. En combinant les capacités de **visualisation rapide de MATLAB** avec la

puissance de calcul de C++, l'étudiant devient capable de développer ses propres outils d'analyse et de modélisation.

Dans le prochain chapitre, nous passerons à l'application directe de ces compétences à travers l'**initiation aux logiciels géotechniques** dédiés comme **Géoslope, Talren, et LimitState-Geo**, avant de nous concentrer sur PLAXIS.