

# Programmation pour Ingénieur

*ME 3<sup>e</sup> semestre*

*rev. 2025.r2*

Christophe Salzmann

Laboratoire  
d'Automatique

# modifications

- 15.08.2025, r1 – version initiale
- 4.09.2025, r2 – update planning

# Plan

- Objectifs
- Compétences
- Motivation
- Organisation
- Contenu
- Notes
- Projet
- AI

# Objectifs

- Appliquer, approfondir et développer les connaissances acquises lors du cours informatique ICC
- Apprendre à écrire un programme, le debugger et le tester
  - i.e. apprendre à se débrouiller tout seul
- Etre à même de passer d'une idée à un programme en choisissant l'environnement le mieux adapté
- Se familiariser avec l'acquisition, le traitement et la visualisation de données

# Motivation – contrôle du babyfoot



Vision: C, Control: LabVIEW, simulation: matlab

# Motivation – projet

*Mener à bien un projet complet*

*Affiner vos connaissances*

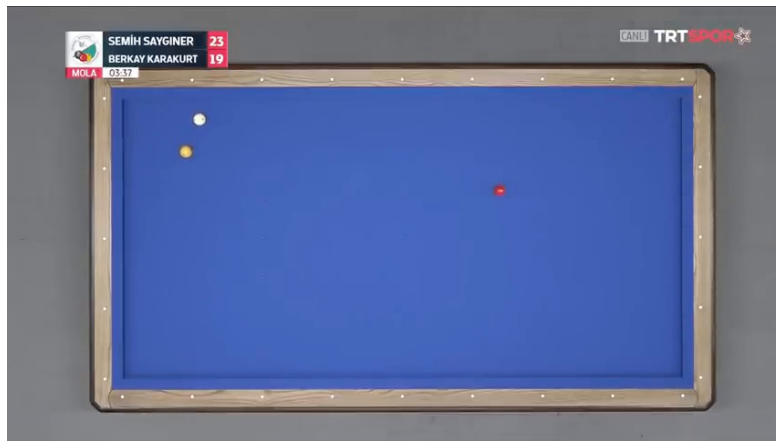
*Apprendre à décomposer un problème en sous tâches*

*Apprendre à échanger de l'information entre applications*

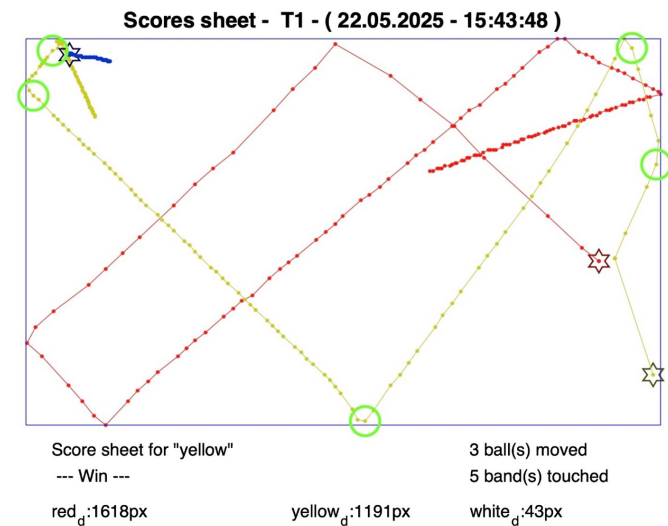
*Apprendre à gérer/debugger vos programmes*

*Apprendre les bonnes pratiques*

*Implémentation à l'aide d'outils performants*



<https://www.youtube.com/watch?v=JVMQzd-1yMg>



# Compétences

## Domaine

- Comprendre les bases de la programmation
- Développer un logiciel (simple) structuré en utilisant un langage/environnement de programmation tel que C, Fortran ou Matlab

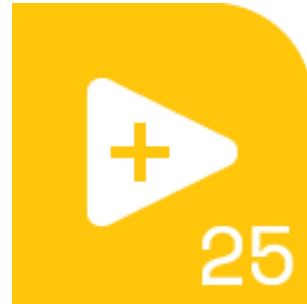
## Transversales

- Gérer son temps
- Montrer de la persévérance en dépit des obstacles

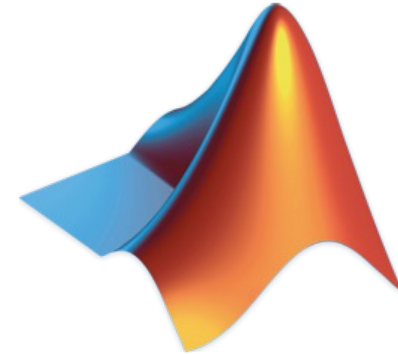
# Motivations



C



LabVIEW



Matlab

- 2 nouveaux langages
- 3 manières différentes d'appréhender un problème
- 3 manières différentes d'exprimer une idée

**3 fois mieux préparé pour les cours à venir!**

# Organisation

- Cours & Exercices
  - jeudi, salle: **CE6**, 14:15-16:00
- Projet
  - lundi, salle: **INF1, INF2 et INF3**, 16:15-18:00
- VM ou votre ordinateur personnel
- Projet (à 3) couvrant les 3 langages
- Evaluations 2 x
- Page Moodle

## SEMAINE DE RÉFÉRENCE

	Lu	Ma	Me	Je	Ve
8-9					
9-10					
10-11					
11-12					
12-13					
13-14					
14-15					
15-16					
16-17	INF1 INF2 INF3				
17-18	INF1 INF2 INF3				
18-19					
19-20					
20-21					
21-22					

■ Cours   ■ Exercice, TP   ■ Projet, autre

<https://moodle.epfl.ch/course/view.php?id=15906>

# Plan des cours 2025 (tentative)

v.5.9.2025

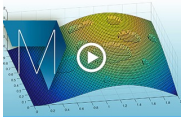
Wk	Matière	Date cours	Date exo/proj	Exercices	Projets
1	Introduction, projet, rappel C	11.9	8.9	Pas de séance Rev. C1 @ home	Pas de séance Rev. C1 @ home
2	Fichiers, représentation des nombres, Bitwise op.	18.9	15.9	C1 Revision	PrjC1 (PM_2025.h)
3	Software Engineering	25.9	22.9	Congé Rev. C1 @ home	Congé Rev. C1 @ home
4	Listes, tri, gestion de la mémoire	2.10	29.9	C2 Fichier	C3 Listel32
5	Matlab I	9.10	6.10	C4 ListePtr (sans q4)	PrjC2
6	Matlab II	16.10	13.10	ML1	PrjC3
	<i>Vacances</i>	<i>23.10</i>	<i>20.10</i>		
7	Matlab III	30.10	27.10	ML2	PrjML1
8	Matlab IV	6.11	3.11	ML3	PrjML2
9	LabVIEW – I	13.11	10.11	ML4	PrjML3
10	LabVIEW – II	20.11	17.11	LV1 my Scnd VI	PrjML4
11	LabVIEW – III	27.11	24.11	LV2 Tunnel-SR	PrjLV1
12	LabVIEW – IV	4.12	1.12	LV3 Design P.	PrjLV2
13	<b>Final</b> (à confirmer)	11.12	8.12	PrjLV3	PrjLV3
14	intro python	18.12	15.12	PrjLV4	PrjLV4
15	<b>Remise projets: (tba)</b>				

# Notes - *tentative*

## Pondération

- Examen final 50%
- Projet 50%

# Références



**MOOC EPFL - introduction à matlab**

**<https://www.edx.org/course/matlab-et-octave-pour-debutants-epfl-matlaboctavex>**



**National Instruments official videos on learning LabVIEW**

**<https://learn.ni.com/learning-paths/labview-core-1>**

# Références



**"Introduction to algorithms / Thomas H. Cormen ... [et al.]".**  
**Année:2009. ISBN:978-0-262-03384-8**



**Algorithmes et structures de données avec ADA, C++ et Java**  
**2004, PPUR, ISBN: 2-88074-488-1**



**Algorithms. R. Sedgwick, edition in c++**



**MATLAB A Practical Introduction to Programming and Problem Solving**  
**Third Edition, Stormy Attaway**  
**<http://www.sciencedirect.com/science/book/9780124058767>**



**Scientific Computing with MATLAB and Octave**  
**Second Edition, Alfio Quarteroni & Fausto Saleri**



**LabVIEW for Everyone: Graphical Programming Made Easy**  
**and Fun (3rd Edition), Jeffrey Travis, Jim Kring**



# Projet

- Le projet se fait par groupe de 3. Les membres du groupe sont responsables de l'entier du code, la note est donnée au groupe, sauf cas exceptionnel.
- Ce projet illustre comment échanger de manière simple des données entre les différents programmes.
- La donnée du projet est volontairement libre sur certains points. Vous aurez la tâche de choisir une solution adaptée et de la mettre en œuvre. Vous devrez documenter votre code.
- Le projet est formaté de telle sorte que sa validation se fera "en 1 click". Différents tests seront réalisés sur vos programmes. La partie LabVIEW contrôle les 2 autres parties.

# Logiciels pour vos laptops

Les logiciels **LabVIEW** et **MATLAB** sont disponibles pour les étudiants via Poseidon. *Il faut installer de préférence les versions **anglaises**!*

*LabVIEW 20**25**Q3*

<https://www.epfl.ch/campus/services/ressources-informatiques/support-informatique/logiciels-pro/>

Les compilateurs C/C++ (gratuits) sont à installer depuis les sites des fabricants.

Visual Studio <https://visualstudio.microsoft.com>

Xcode via l'AppStore <https://apps.apple.com/app/xcode/id497799835>

**~~Δ Visual Studio ≠ Visual Studio Code!~~**

# Resources en ligne

**MATLAB Online** (uniquement avec email EPFL)

<https://matlab.mathworks.com>

**Sysquake mobile** (free)

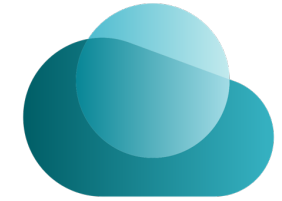
<https://calerga.com/SQjs/index.html>

**Compilateurs C/C++**

<https://www.onlinegdb.com>

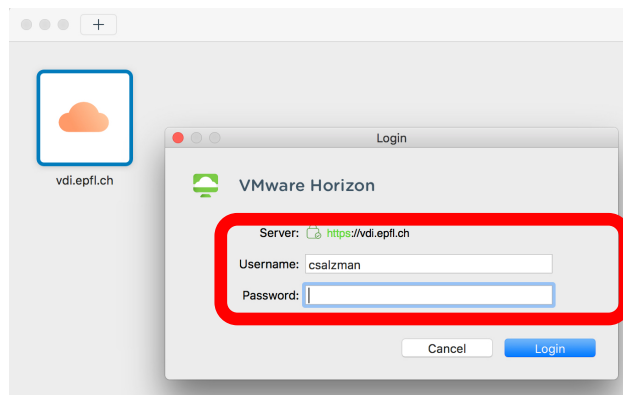
[https://www.tutorialspoint.com/compile\\_c99\\_online.php](https://www.tutorialspoint.com/compile_c99_online.php)

# VM – STI\_Windows11

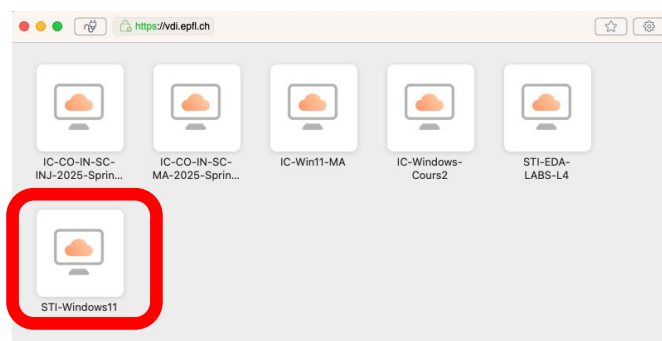


Vous avez la possibilité d'employer une machine virtuelle (VM) spécialement configurée pour ce cours: Visual studio (compilateur C/C++), LabVIEW, Matlab sont installés, à jour et prêts à l'emplois.

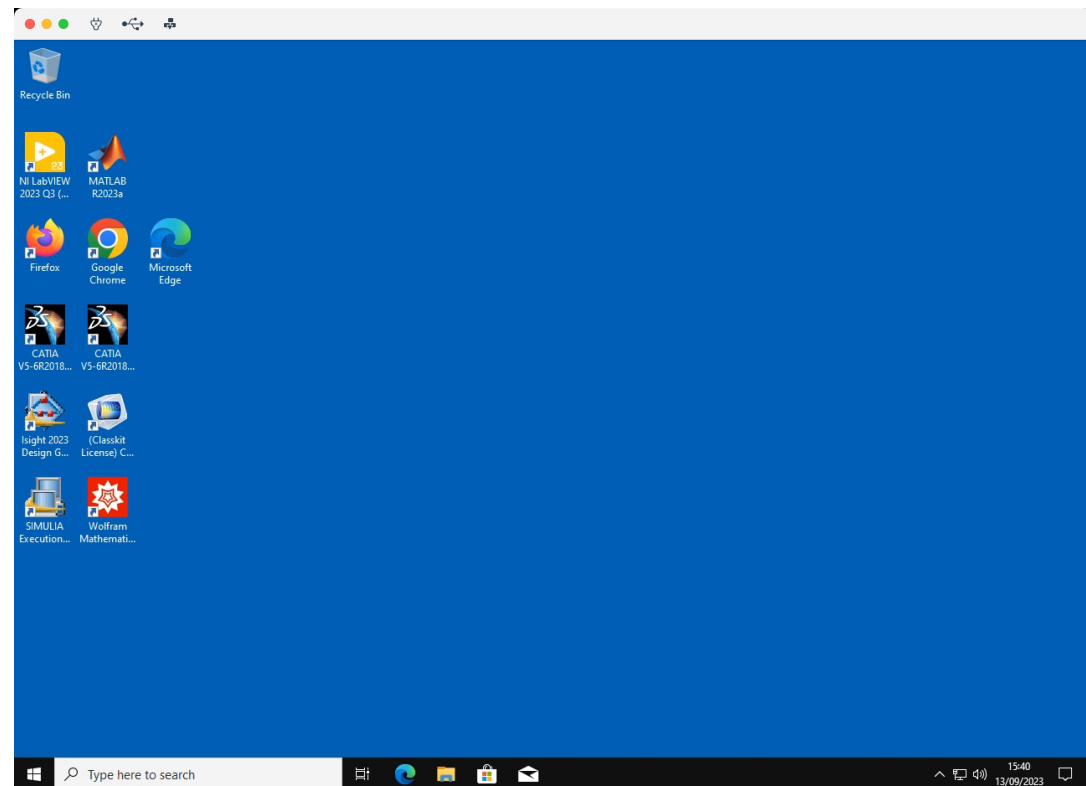
Utilisez VPN si nécessaire



**Camipro credentials**



**STI\_Windows11**



# ***AI and the like***

- Vous pouvez utiliser **intelligemment** des outils AI
- Pour le projet, vous devez me remettre le *transcript* au format PDF de vos discussions le concernant
- Soyez très critique avec les réponses fournies
- ChatGPT et autres outils similaires ne sont ni des compilateurs, ni des débbugger, ils prédisent la suite du texte en se basant sur ce qu'il a appris durant leur entraînement.
- Est-ce que programmer à encore un sens ?
  - Vous avez appris les 4 opérations mathématique de base et vos tables de multiplications alors que des machines à calculer le font mieux et plus vite que vous.
- Puis-je faire tout le projet avec l'IA ?

# Discussion avec ChatGPT 4

Un étudiant a soumis le bout de code suivant "in A[10] = {-1};" que peux tu m'en dire?

[skip]

contient **plusieurs erreurs**, à la fois **de syntaxe** et **de sémantique** en langage C (ou C++).

[skip]

## Interprétation pédagogique :

Si l'étudiant écrit in A[10] = {-1};, cela peut indiquer :

Un **copier-coller maladroit** (probablement depuis ChatGPT ou autre outil).

Une **compréhension superficielle** de la syntaxe C.

Une **influence possible** d'autres langages (par exemple, Python ou pseudocode).

exacte c'est un copier-coller depuis ChatGPT. comment est-il possible que du code erroné soit donné en réponse ?

## Pourquoi ChatGPT peut produire du code erroné ?

### 1. ChatGPT n'exécute pas le code — il le prédit

ChatGPT **ne compile ni n'exécute** le code qu'il produit.

Il **prédit la suite de texte la plus probable** en se basant sur ce qu'il a "vu" durant son entraînement.

Si dans les données d'entraînement, une mauvaise syntaxe a été vue dans un contexte similaire, il peut la reproduire.

### 2. Contexte imprécis ou ambigu

Si l'utilisateur fournit **peu ou pas de contexte** (par exemple : "déclare un tableau A avec -1 comme valeur initiale"), ChatGPT peut mal interpréter ce que vous voulez faire.

Il pourrait alors **générer une réponse syntaxiquement incorrecte** selon le langage attendu.

# Discussion avec ChatGPT 4

## 3. Influence d'autres langages ou pseudocode

ChatGPT a été exposé à **de nombreux langages de programmation** et même à du pseudocode.

Il peut **confondre** ou **mélanger** des syntaxes entre C, Python, Pascal, ou pseudocode — surtout dans des cas simples comme des déclarations de tableau.

## 4. Erreur d'édition ou de troncation

Parfois, ChatGPT génère un code correct **dans une première version**, mais une reformulation ou un raccourci **tronque ou déforme la syntaxe** (comme écrire in au lieu de int).

Cela arrive quand il essaie de **“faire joli”** ou résumer une idée.

[skip]

## En résumé :

ChatGPT peut produire du code erroné car il ne raisonne pas comme un compilateur ou un développeur, mais **comme un modèle statistique de langage**. Il est très bon pour suggérer, mais pas infallible — surtout quand il est utilisé sans esprit critique.

[skip]