

Algorithmique et langage C

— Environnement et Outils pour ce Cours —



CERI Systèmes Numériques

Luc Fabresse

luc.fabresse@imt-nord-europe.fr

MyLearningSpace (MLS)

<https://mylearningspace.imt-nord-europe.fr/course/view.php?id=143>

Tout est sur MLS.

Si question / problème, demander sur discord.

Pour accéder à discord, cf. MLS.

Hardware

- Pc ou mac
- Connexion internet
- Un casque micro (pour discord si besoin)

Software

- Discord <https://discord.com/>
- git <https://git-scm.com/downloads>
- vscode <https://code.visualstudio.com/>
- compte github <https://github.com>
- compilateur C (gcc, clang ou mingw pour Windows)



Installer Linux

- Installer Ubuntu <https://ubuntu.com/download/desktop> ou un autre linux
- Installer tous les logiciels requis pour compiler du C

```
sudo apt-get update  
sudo apt-get install -y cmake gcc gdb gdbserver clang lldb \  
build-essential valgrind git
```



Une solution si vous ne pouvez ou voulez pas installer Linux.

Docker

- Installer l'extension remote container de VSCode
[https ://code.visualstudio.com/docs/remote/containers](https://code.visualstudio.com/docs/remote/containers)
- Installer Docker [https ://docs.docker.com/get-docker/](https://docs.docker.com/get-docker/)
- Forker le dépôt [https ://github.com/ceri-num/uv-mlod-vscode-template](https://github.com/ceri-num/uv-mlod-vscode-template)
- Clone votre dépôt sur votre machine
- Ouvrir le projet avec VSCode - remote container

Le prof ne peut pas trop aider car il n'a pas Windows.

A faire :

- Installer git for Windows <https://gitforwindows.org/>
- Installer Mingw <https://sourceforge.net/projects/mingw/> Tuto installation avec VSCode : <https://code.visualstudio.com/docs/languages/cpp>
- Configurer la variable d'environnement PATH pour que la commande gcc soit correctement trouvée



Comment travailler avec git dans ce cours ?

- ❶ Créer votre compte github <https://github.com/bobleponge>
- ❷ **Forker** mon dépôt sur votre compte github
<https://github.com/LucFabresse/mlod-algoc-sujets>
vers <https://github.com/bobleponge/algoc>
- ❸ **Cloner** votre dépôt sur votre PC
<https://github.com/bobleponge/algoc>
- ❹ Vous pouvez travailler : `commit`, `pull`, `push`

Récupérer les mises à jour du cours !

<https://docs.github.com/en/pull-requests/collaborating-with-pull-requests/working-with-forks/syncing-a-fork>

A faire une seule fois

Ajouter un dépôt upstream à votre clone

```
git remote add upstream https://github.com/LucFabresse/mlod-algoc-sujets
```

A faire régulièrement

```
git fetch upstream
```

```
git merge upstream/main
```



De nos jours, il y a beaucoup de ressources :

- Web
 - <https://roadmap.sh/>
 - <https://github.com/jwasham/coding-interview-university>
 - ...
- IA (ChatGPT, Copilot, ...)
- ...

Qu'attendez-vous d'un prof ?

- **Je** vous propose un parcours (sujets) i.e. quelles notions comprendre, quels exercices pour les comprendre, ...
- **Vous** travaillez
- **Je** vous accompagne : regarde votre travail, vous propose des améliorations, réponds à vos questions, ...

Je n'ai pas pour but d'éclairer ceux qui ne sont pas désireux d'apprendre, ni éveiller ceux qui ne sont pas soucieux de donner une explication eux-mêmes. Si je leur ai montré un angle du carré et qu'ils ne peuvent pas revenir à moi avec les trois autres, je ne devrais pas revenir sur le premier angle.

[Confucius]

- je lis les cours et prépare des questions avant les séances
- j'arrive à l'heure en cours / TP
- je suis attentif
- je pose des questions
- je réponds aux questions des autres si j'ai compris
- je termine / paufine / revois ou m'avance en dehors des séances
- je travaille régulièrement
- je rends tous les travaux demandés dans les temps

Ne pas comprendre des choses est normal. C'est votre attitude face à cela qui déterminera si vous échouerez.