

Sigle : INF1633 Gr. 01**Titre : Programmation de systèmes embarqués en C/C++****Session : Automne 2021 Horaire et local****Professeur : Adi, Kamel****1. Description du cours paraissant à l'annuaire :****Objectifs**

Permettre à l'étudiant(e) de maîtriser, par la pratique, la programmation en langage C/C++ des systèmes informatiques embarqués.

Contenu

Introduction au domaine des systèmes embarqués. Aspects matériels : architecture RISC et microcontrôleurs ARM et ATMEGA. Langage C/C++. Chaîne de compilation GNU. Environnements de programmation. Développement et intégration d'applications dans des environnements embarqués. Techniques de débogage. Introduction aux systèmes d'exploitation temps réel (RTOS). Réalisation d'un projet de système embarqué. Ce cours comporte des séances obligatoires de travaux dirigés (TD) de deux heures par semaine.

[Descriptif – Annuaire](#)

2. Objectifs spécifiques du cours :

Permettre d'acquérir les compétences nécessaires afin de devenir un bon programmeur C/C++ sur des plateformes embarquées :

- Écrire du code bien structuré et bien optimisé
- Savoir documenter le code
- Savoir manipuler les bibliothèques
- Maîtriser un environnement de programmation pour micro-contrôleurs

3. Stratégies pédagogiques :

Le cours se donne dans une salle de laboratoire. Les séances de cours seront présentées sous forme magistrales parsemées d'exercices pratiques. Le matériel pédagogique est accessible à partir de la plateforme Moodle dédiée au cours. Un forum de discussion sera aussi disponible pour poser des questions liées à la matière enseignée.

Des travaux dirigés et pratiques, sous la supervision d'un assistant d'enseignement, seront également réalisés afin de consolider les concepts présentés durant les séances de cours.

4. Heures de disponibilité ou modalités pour rendez-vous :

Communication par courriel (Kamel.Adi@uqo.ca) et via le forum de discussion.

5. Plan détaillé du cours sur 15 semaines :

Semaine	Thèmes	Dates
1	Présentation du cours Introduction à la programmation embarquée et aux langages C/C++	09 sept. 2021

2	Introduction aux micro-contrôleurs Kinetis et à l'architecture ARM Présentation de la carte Freescale FRDM-KL27z	16 sept. 2021
3	Langage C <ul style="list-style-type: none"> • Structure d'un programme • Éléments de base • Les entrées/sorties Travail pratique 1	23 sept. 2021
4	Environnements de programmation <ul style="list-style-type: none"> • Mbed • MCUXpresso Techniques de débogage Travail pratique 2	30 sept. 2021
5	Langage C <ul style="list-style-type: none"> • Pointeurs, adressage et gestion de la mémoire • Fonctions, passage de paramètres • Récursivité, fonctions récursives 	07 oct. 2021
6	Semaine d'études	14 oct. 2021
7	Éléments avancés du langage C Structures de données en C	21 oct. 2021
8	Examen de mi-session	28 oct. 2021
9	Système d'interruptions et programmation multitâches Publication de l'énoncé pour le travail de session	04 nov. 2021
10	Communications dans l'Internet des objets <ul style="list-style-type: none"> • La norme Zigbee • La carte Xbee Travail pratique 3	11 nov. 2021
11	Programmation orientée objets en C++ <ul style="list-style-type: none"> • C++ versus C • Classes, méthodes et attribues • Exemple de code C++ Travail pratique 4	18 nov. 2021

12	Langage C++ <ul style="list-style-type: none"> • L'encapsulation de données, la surcharge d'opérateurs et de méthodes • L'héritage • Le polymorphisme Travail pratique 5	25 nov. 2021
13	Systèmes d'exploitation temps réels FreeRTOS	02 déc. 2021
14	Présentation des projets de session	09 déc. 2021
15	Examen final	16 déc. 2021

6. Évaluation du cours :

Dans le cas spécifique du cours **Programmation de systèmes embarqués en C/C++**, l'attribution des notes se fera selon la répartition suivante :

- **Examen de mi-session : 30 %**
- **Examen final : 40 %**
- **Projet de session : 30 %**

Une moyenne inférieure à 50 % aux examens est éliminatoire et conduit automatiquement à l'échec.

7. Politiques départementales et institutionnelles :

- Politique du département d'informatique et d'ingénierie relative à la tenue des examens
- Note sur le plagiat et sur la fraude
- Politique relative à la qualité de l'expression française écrite chez les étudiants et les étudiantes de premier cycle à l'UQO
- Absence aux examens : cadre de gestion, demande de reprise d'examen (formulaire)

À l'UQO, **les violences à caractère sexuel, c'est tolérance zéro!**

La communauté universitaire s'engage à lutter contre les inconduites, le harcèlement et les violences à caractère sexuel : parce que **le respect, c'est l'affaire de tout le monde!**

N'oubliez pas de faire la formation obligatoire :

uqo.ca/bimi/formation-obligatoire

Pour de plus amples renseignements :

bimi@uqo.ca



8. Principales références :

- Documents sur le site du cours Moodle.
- **ErychStyger**, Blog at <https://mcuoneclipse.com/>, **Freescale/NXP**, Lucerne University of Applied Sciences and Arts.
- Kinesis FDRM-KL26Z and Software Development Kit: documentation, tools and guides, NXP.
- K. Hong. C++ Tutorial Home. <http://www.bogotobogo.com/cplusplus/cpptut.php>.
- Michael Barr, Anthony Massa, "Programming Embedded Systems, With C and GNU Development Tools", O'Reilly Media, 2009.
- Christopher Blaess. Solution temps-réel sous Linux. **Cas pratique : le Raspberry Pi. Avec 50 exercices corrigés**

2^e édition, Eyrolles, 2012.

- FreeRTOS, [FreeRTOS Books and Manuals](http://www.freertos.org/), at <http://www.freertos.org/>.
- Jean LABROSSE, "*MicroC/OS-II The Real-Time Kernel*", *MicriumuC/O-III* for the Freescale kinetis, 2011.
- POSIX Threads Programming: <https://computing.11nl.gov/tutorials/pthreads/>.

9. Page Web du cours :

<https://moodle.uqo.ca>