Sigle : INF1633 Gr. 20**Titre : Programmation de systèmes embarqués en C/C++****Session : Automne 2024 Horaire et local****Professeure : Bouhaddi, Myria****1. Description du cours paraissant à l'annuaire :****Objectifs**

Au terme de ce cours l'étudiant.e aura maîtrisé, 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).

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 microcontrôleurs

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 à faire à la maison, et travaux pratiques obligatoires et en mode présentiel, 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 (Myria.Bouhaddi@uqo.ca) et via le forum de discussion.

5. Plan détaillé du cours sur 15 semaines :

Semaine	Thèmes	Dates
1	Introduction au cours et aux systèmes embarqués <ul style="list-style-type: none"> • Présentation du cours, objectifs et méthodologie. • Vue d'ensemble des systèmes embarqués et des langages C/C++. 	05 sept. 2024
2	Découverte des microcontrôleurs et de l'architecture ARM <ul style="list-style-type: none"> • Introduction aux microcontrôleurs Kinetis et à l'architecture ARM. • Présentation pratique de la carte Freescale FRDM-KL27z. 	12 sept. 2024

3	Fondamentaux du langage C <ul style="list-style-type: none"> Structures de programmes, éléments de base, entrées/sorties. Travail pratique 1: 24 septembre	19 sept. 2024
4	Environnements de programmation et techniques de débogage <ul style="list-style-type: none"> Introduction aux environnements Mbed et MCUXpresso. Techniques de débogage appliquées aux systèmes embarqués. Travail pratique 2 : 02 octobre	26 sept. 2024
5	Langage C <ul style="list-style-type: none"> Pointeurs, adressage et gestion de la mémoire. Fonctions, passage de paramètres, récursivité. Travail pratique 3 : 09 octobre	03 oct. 2024
6	Approfondissement en C <ul style="list-style-type: none"> Structures de données en C Publication de l'énoncé pour le travail de session	10 oct. 2024
7	Semaine d'études	17 oct. 2024
8	Examen de mi-session	24 oct. 2024
9	Système d'interruptions et programmation multitâches <ul style="list-style-type: none"> Exploration des interruptions et de la programmation multitâches en C/C++. 	31 oct. 2024
10	Communications dans l'Internet des objets <ul style="list-style-type: none"> Introduction à la norme Zigbee et à la carte Xbee. Applications pratiques de l'IoT dans les systèmes embarqués. Travail pratique 4 : 13 novembre	07 nov. 2024
11	Introduction au C++ et concepts de base <ul style="list-style-type: none"> Transition de C à C++ : différences clés, gestion de la mémoire et types de données. Syntaxe de base en C++, discussion sur la standardisation et la portabilité. Travail pratique 5 : 20 novembre	14 nov. 2024
12	Avancé en C++ <ul style="list-style-type: none"> Encapsulation de données, surcharge d'opérateurs et de méthodes. Héritage et polymorphisme pour une conception modulaire. 	21 nov. 2024
13	Systèmes d'exploitation temps réels (RTOS) <ul style="list-style-type: none"> Introduction à FreeRTOS et à ses applications dans les systèmes embarqués. 	28 nov. 2024
14	Présentation et discussion des projets de fin de session	05 déc. 2024
15	Examen final	12 déc. 2024

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 :

- Politiques relatives à la tenue des examens
- Note sur le plagiat et les fraudes
- 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)

Tolérance **ZÉRO** en matière de violence à caractère sexuel.

Le Bureau d'intervention et de prévention en matière de harcèlement (BIPH) a pour mission d'accueillir, soutenir et guider toute personne vivant une situation de harcèlement, de discrimination ou de violence à caractère sexuel. Le BIPH oriente ses actions afin de prévenir les violences à caractère sexuel pour que nous puissions étudier, travailler et s'épanouir dans un milieu sain et sécuritaire.

Vous vivez ou êtes une personne témoin d'une situation de violence à caractère sexuel ? Vous êtes une personne membre de la communauté étudiante ou une personne membre du personnel, autant à Gatineau qu'à Ripon et St-Jérôme, l'équipe du BIPH est là pour vous, sans jugement et en toute confidentialité.

Ensemble, participons à une culture de respect.

Pour de plus amples renseignements consultez UQO.ca/biph ou écrivez-nous au Biph@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 Developmet 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, 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>